ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ

Кваліфікаційна наукова
праця на правах рукопису

ЗАГОРОДНЯ Діана Іванівна

УДК 004.93

ДИСЕРТАЦІЯ

ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ СТРУКТУРНО-СТАТИСТИЧНОЇ ІДЕНТИФІКАЦІЇ ІЄРАРХІЧНИХ ОБ'ЄКТІВ ЗА ХАРАКТЕРНИМИ ТОЧКАМИ ЇХ КОНТУРІВ В АВТОМАТИЗОВАНИХ СИСТЕМАХ ВІДЕОНАГЛЯДУ

05.13.06 – Інформаційні технології

05 «Технічні науки»
(галузь знань)

Подається на здобуття наукового ступеня кандидата технічних наук

Дисертація містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

Загородня Д.І.
(підпис, ініціали та прізвище здобувача)

Наукові керівники (консультанти): доктор технічних наук, професор
Крилов Віктор Миколайович,
доктор, професор
Уве Гросман (Uwe Großmann)

Ідентичність всіх примірників дисертації
ЗАСВІДЧУЮ:
Вчений секретар спеціалізованої
вченої ради
/В.В. Яцків/

Тернопіль – 2017

АНОТАЦІЯ

Загородня Д.І. Інформаційна технологія структурно-статистичної ідентифікації ієрархічних об’єктів за характерними точками їх контурів в автоматизованих системах відеонагляду. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня кандидата технічних наук за спеціальністю 05.13.06 інформаційні технології. – Тернопільський національний економічний університет МОН України, Тернопіль, 2017.

Дисертація присвячена розробленню методів контурної сегментації на основі методу Канні та виділенню характерних точок контуру для автоматизованих систем відеонагляду для підвищення швидкодії при обмежених обчислювальних ресурсах.

У дисертаційній роботі в першому розділі проаналізовані існуючі автоматизовані системи відео нагляду (АСВН), в яких кількість камер постійно зростає і, відповідно збільшується ресурсоємність. Проте суттєвим обмеженням таких систем є потреба великої кількості операторів для їх обслуговування. Це зумовило перехід до інтелектуальних АСВН. Проте, такі системи знижують оперативність АСВН і потребують значних обчислювальних ресурсів. Для підвищення оперативності АСВН запропоновано зменшувати обсяг даних, що обробляються на етапах сегментації та ідентифікації об’єктів розпізнавання використовуючи методи вейвлет-аналізу. Методи вейвлет-аналізу дозволяють виділяти на відеоінформації об’єкти і деталі об’єктів різних геометричних розмірів, відповідно до заданої цілі обробки та використовувати ієрархічні методи та базові інформаційні технології (ІТ) контурної сегментації (КС), виділення характерних точок (ХТ) та ідентифікації, що в свою чергу дозволяють підвищити оперативність прикладних інтелектуальних інформаційних систем відеонагляду при достатньому рівні достовірності розпізнавання.

У другому розділі отримав подальший розвиток метод та базова ІТ контурної сегментації на базі методу Канні. Цей метод полягає у : згортці рядка

зображення вейвлет-функцією (окремо по рядках та по стовпцях); пороговому фільтрі (встановленні порогу встановлюється для значення інтенсивності пікселів: якщо значення інтенсивності менші за поріг, то заміняємо її на 0, ту ж, що вища порогу – залишаємо без змін); обчисленні величини градієнта для визначення величини стрибка інтенсивності та його напряму, а в кінці отримані контурні препарати складаємо за схемою логічного «або».

Після того, як отримали перший контур (найвищого рівня ієрархії), для отримання контуру наступного рівня ієрархії виконуються ті самі дії із застосуванням вейвлета меншої довжини та обробляється не вся відеоінформація, а її частина, що знаходиться всередині контуру попереднього рівня ієрархії.

Для згортки рядка зображення запропоновано використати вейвлет Хаара, який реалізує множення на «+1» і «-1», що зночно спрощує обчислення і, тим самим, підвищує оперативність.

У виборі довжини вейвлету керуються геометричними розмірами об‘єктів як складових частин відеоінформації. На першому етапі довжина вейвлету повинна відповідати геометричним розмірам «найбільшого об’єкта» і з переходом на наступні рівні повинна зменшуватись відповідно до розмірів об’єкта. Критерієм припинення деталізації буде досягнення такого рівня деталізації, що об’єкти які цікавлять, будуть виділеними.

Перевагою розробленого методу є замкнутість контурів, а також можливість ігнорування надлишковою відеоінформацією за допомогою згортки відеоінформації вейвлет-функцією. Це дозволяє розбити відеоінформацію на сукупність ієрархічних контурів деталізації («контур – підконтур»). А експериментальні дослідження показали високу завадостійкість запропонованого методу за критерієм Прета (у 1,2 – 1,5 рази вищу у порівнянні з методом Канні).

Розроблено метод та базову інформаційну технологію (ІТ) виділення характерних точок контуру з регульованою деталізацією об’єктів, в основі яких лежить: формування функції кривизни, після цього здійснюється вейвлет-перетворення функції кривизни в базисі Хаара і до ХТ відносять ті точки, які знаходяться на перетині графіка функції з віссю ох. Якщо рівень деталізації не задовольняє, то зменшуємо довжину вейвлета та повторюємо описану процедуру. Експериментальні дослідження показали, що розроблений метод та базова інформаційна технологія суттєво перевершує диференціальний метод по завадостійкості у 1,2 – 2 рази, а інтерполяційний метод по точності у 1,5 – 3 разів.

У третьому розділі удосконалено метод та базову ІТ структурно-статистичної ідентифікації об’єктів на базі застосування ієрархічної послідовності ідентифікаційних векторів, розрахованих на основі характерних точок контуру об’єкта з регульованою деталізацією. Даний метод забезпечує інваріантність до геометричних перетворень об’єктів (зсуву в полі зору, повороту об’єкта та масштабування), володіє високою завадостійкістю.

В результаті контурного аналізу відеоінформації на базі запропонованого підходу отримано пірамідальне представлення геометричних форм об’єкта і його деталей: набір контурних препаратів. Таке представлення використовується для обчислення геометричних ідентифікаційних ознак об’єкта. Даний підхід структурно-ієрархічної контурної сегментації дозволяє підвищити швидкодію роботи системи відеоспостереження за рахунок зменшення об’єму інформації, необхідної для обробки.

Структурне представлення зображення у вигляді сукупності примітивних геометричних елементів (в нашому випадку сукупності контурів) і їх відношення між собою, побудоване на базі геометричного підходу, включаючи в опис форми взаємне розташування і відношення окремих елементів. Тобто, зображення можна представити у вигляді сукупності контурів різного рівня ієрархії.

Суть методу полягає в застосуванні ієрархічної послідовності ідентифікаційних векторів, розрахованих на основі інформації про ХТ контуру об’єкта з регульованою деталізацією.

Після того, як зображення представлено у вигляді сукупності контурів та їх відношень між собою з різним рівнем ієрархії , відбувається виділення ХТ контуру (спочатку на контурі найвищого рівня ієрархії), побудова ідентифікаційного вектора на основі інформації про ХТ контуру, проведення класифікації на відповідному рівні та віднесення об'єкта до відповідного класу. Після цього відбувається перехід на нижчий рівень ієрархії та проводиться розпізнавання всередині класу, в який потрапив об'єкт. Ця процедура триває доти, поки не прийнято остаточного класифікаційного рішення.

Ідентифікаційний вектор обчислюється на основі методу геометричних моментів-ознак з використанням інформації виключно про характерні точки контуру. До переваг цього методу відносять інваріантність до паралельного переносу, стійкість до зміни масштабу, інваріантність до кута повороту об'єкта, високу стійкість методу, ознаки легко нарощуються до отримання стійкого розпізнавання, а мала кількість ХТ визначає високу обчислювальну ефективність методу.

На основі розроблених методів у четвертому розділі створено ІТ структурно-статистичної ідентифікації ієрархічних об'єктів за ХТ контурів на основі створених методів та базових ІТ контурної сегментації на базі методу Канні виділення ХТ контуру з регульованою деталізацією, яка використовується у комп'ютерній системі відеонагляду.

Ідентифікація об'єктів за обличчями проводиться на декількох рівнях: на першому рівні об'єкт відноситься до певного класу за формою голови. Якщо даної інформації не достатньо, то здійснювався перехід на наступний рівень ієрархії об'єкту (ідентифікаційні вектори контурів носа, очей та рота).

На базі запропонованих у розділах 2 і 3 методів, в рамках науково-дослідної роботи № 0112U008453 було розроблено АСВН, яку програмно реалізовано в середовищі Visual Studio 2015 на мові програмування C# з використанням бібліотек OpenCV.

Розроблені ІТ впроваджено у АСВН у ПП «ЕКОДЕРІ» , ТОВ “ВІП комп’ютер”, ТзОВ завод «РЕМПОБУТТЕХНІКА», а також використано у навчальному процесі.

Дослідження показали, що застосування цих процедур в інформаційній технології відеонагляду дозволило зменшити час обробки кадру у 8-15 разів (в залежності від вмісту кадру та кількості об’єктів на ньому) при невеликому зниженні достовірності (5-10%)

Ключові слова: автоматизована система відеонагляду, контур, вейвлет-аналіз функції кривизни, характерні точки, структурно-статистична ідентифікація, ідентифікаційний вектор.

Список публікацій здобувача

Статті в наукових фахових виданнях та закордонних журналах:

1. Загородня Д.І. Дослідження диференціального методу виділення характерних точок контуру облич / Д.І. Загородня. В.І. Дорош, Н.С. Добровольська, О.Л. Римар // Міжнародний науково-технічний журнал «Вимірювальна та обчислювальна техніка в технологічних процесах». – Хмельницький, 2014. – №4 (49). – С. 162-166.

2. Загородняя Д.И. Метод идентификации лиц по характерным точкам контура / Д.И. Загородняя // Вестник Брестского государственного технического университета. Серия физика, математика, информатика. – 2015. – №5. – С. 30-33.

3. Загородня Д.І. Аналіз інтерполяційного методу виділення характерних точок контуру / Д.І. Загородня // Науковий вісник Чернівецького національного університету ім. Ю. Федьковича. Серія: Комп’ютерні системи та компоненти. – 2015. – Том 6, Випуск 2. – С. 88-92.

4. Загородня Д.І. Порівняння методів виділення характерних точок контуру / Д.І. Загородня // Радіоелектронні і комп’ютерні системи. Харків «ХАІ». – 2016. – № 3 (77). – С. 56-61.

5. Zahorodnia D.I. Canny-based Method of Image Contour Segmentation / Diana Zahorodnia, Yuriy Pigovsky, Pavlo Bykovyy // International Journal of Computing. Ternopil: TNEU. – 2016. – Vol. 15. – Issue 3. – P. 200-205. (*включено до науко-метричної бази Scopus*).

6. Загородня Д.І. Метод виділення контурів зображень з використанням вейвлет-перетворень / Д.І. Загородня, У. Гросман , В.М. Крилов, В.І. Дорош // Науковий вісник Чернівецького національного університету ім. Ю. Федьковича. Серія: Комп'ютерні системи та компоненти. –2016. – Том 7, Випуск 2. – С. 136-141.

Опубліковані праці апробаційного характеру:

7. Paliy Ihor. Fast and Robust Face Detection and Tracking Framework / Ihor Paliy, Volodymyr Dovgan, Ognian Boumbarov, Stanislav Panev, Anatoly Sachenko, Yuriy Kurylyak, Diana Zagorodnya // Proceedings of the IEEE 6th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS'2011). – Prague (Czech Republic), September 15-17, 2011. – Vol.1. – P. 430-433. (*включено до науко-метричної бази Scopus*)

8. Zahorodnia Diana. Structural Statistic Method Identifying Facial Images by Countour Characteristic Points / Diana Zahorodnia, Yuriy Pigovsky, Pavlo Bykovyy, Viktor Krylov, Ihor Paliy, Igor Dobrotvor // Proceedings of the 8th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS'2015). – Warsaw (Poland), September 24-26, 2015. – P. 293-297. (*включено до науко-метричної бази Scopus та Web of Science*)

9. Базелюк Д.В. Підхід до розпізнавання емоцій на основі зображення обличчя. / Д.В. Базелюк, Д.І. Загородня, І.О. Палій // Труди XIII міжнародної науково-практичної конференції «Сучасні інформаційні та електронні технології» («СІЕТ-2012»). – Одеса, 4-8 червня 2012. – С. 118.

10. Загородня Д.І. Алгоритми прослідковування контурів та виділення характерних точок в процедурі сегментації системи комп'ютерного розпізнавання / Д.І. Загородня, І.О. Палій, В.М. Крилов // Сучасні комп'ютерні інформаційні технології: Матеріали III Всеукраїнської школи-семінару молодих вчених і студентів (ACIT'2013). – Тернопіль: ТНЕУ, 17-18 травня 2013. – С.129-130.

11. Загородня Д.І. Метод розпізнавання облич за характерними точками контуру в комп'ютерних системах відеоспостереження. / Д.І. Загородня, І.О. Палій, В.М. Крилов // Труди XIV міжнародної науково-практичної конференції «Сучасні інформаційні та електронні технології» («СІЕТ-2013»), 27-31 травня 2013. – Одеса. – С. 49-52.

12. Загородня Д.І. Алгоритм контурної сегментації в системах відеоспостереження / Д.І. Загородня, К.Ю. Ковалок, А.О. Саченко, В.М. Крилов // Всеукраїнська наукова Інтернет-конференція «Економічна кібернетика – інженерія економіки (випуск 24)» / Збірник тез доповідей: випуск 24. – Тернопіль: Тайп, 29-30 листопада 2013. – С. 3-5.

13. Zahorodnia Diana. Contour Segmentation Method in Video Surveillance Systems / Diana Zahorodnia, Kostantin Kovalok, Anatoly Sachenko, Viktor Krylov, Sergiy Nychyporuk // Proceedings of the International Conference "Modern Problems of Radio Engineering, Telecommunications and Computer Science" (TCSET'2014). – Lviv-Slavske (Ukraine), February 25 – March 1, 2014. – P. 405.

14. Загородня Д.І. Налаштування параметрів алгоритму виділення характерних точок на основі вейвлету Хаара / Д.І. Загородня // Сучасні комп'ютерні інформаційні технології: матеріали V Всеукраїнської школи-семінару молодих вчених і студентів ACIT'2015. – Тернопіль: ТНЕУ, 22-23 травня 2015. – С. 68-70.

15. Загородня Д.І. Метод виділення характерних точок зображення на основі вейвлет-аналізу функції кривизни / Д.І. Загородня, К.Ю. Ковалок, В.М. Крилов // Труди XVI міжнародної науково-практичної конференції «Сучасні

інформаційні та електронні технології» («CIET-2015»), 25-29 травня 2015. – Одеса. – С. 50-51.

16. Загородня Д.І. Підхід до структурно-ієрархічної контурної сегментації / Д.І. Загородня // Інформатика, управління та штучний інтелект. Матеріали другої науково-технічної конференції студентів, магістрів та аспірантів. – Харків: НТУ «ХПІ», 26-27 листопада 2015. – С. 31.

17. Zahorodnia D.I. Structural-Hierarchical Principle of Contour Segmentation / D.I. Zahorodnia, P.E. Bykovyy // Current Issues in Modern Technologies: Book of abstracts of the IV International Scientific and Technical Conference of Young Researchers and Students. – Ternopil: TNTU, 25-26 November 2015. – Vol. 2. – P. 20-21.

18. Загородня Д.І. Контроль доступу до комп'ютера на основі розпізнавання зображення обличчя користувача / Д.І. Загородня, О.І. Губчакевич, В.В. Кисіль. // Актуальні задачі сучасних технологій: збірник тез доповідей V міжнародної науково-технічної конференції молодих учених та студентів,– Тернопіль: ТНТУ, 17–18 листопада 2016. – Том 2. – С. 42-43.

ANNOTATION

Zahorodnia Diana, Informational Technology for Structural and Statistic Identification of Hierarchical Objects from Characteristic Points of their Contours in Automated Video Surveillance Systems. – Manuscript.

Thesis for a Ph.D. (candidate of technical science) degree by specialty 05.13.06 - Information technologies. - Ternopil National Economic University, Ternopil, 2017.

Dissertation is dedicated to the contour segmentation methods development on the basis of Canny's method and selection of the contour inflection points for the automated video surveillance systems in order to rise performance speed with limited computing resources.

Existing AVSS were analyzed in the first chapter of the dissertation paper, where the number of cameras is constantly getting bigger and subsequently increases the workload. But as a substantial limitation for such systems is the need of a huge

amount of operators for their service. This caused the switch to the intelligent AVSS. However, such systems decrease the efficiency of the AVSS and require substantial computing resources. To increase efficiency of the AVSS it was proposed to decrease data amount being processed on the segmentation stages and to identify objects for selection using wavelet analysis methods. Wavelet analysis methods allow to select on the video information objects and details of objects with different geometric sizes, corresponding to the given processing aim and to use hierarchical methods and basic information technology of the contour segmentation, selection of the inflection points and identification which in their turn allow to increase productivity of the applied intellectual information systems of the video surveillance with the sufficient level of recognition authenticity.

In the second chapter a further development was given to the method and basic informational technology of the contour segmentation on the basis of Kanny's method. This method consists of convolution of the image row by the wavelet function (separately along the rows and columns); threshold filter (setting a threshold for the value of the pixel intensity: if the intensity value is lower than the threshold, then it is changed into 0 and the one that is higher is left unchanged); calculation of the gradient values to define the size of the intensity jump and its direction and finally obtaining contour items put together with the help of logical "or" scheme.

After the first contour was received (the one from the highest hierarchical level), to receive the contour of the next hierarchical level the same actions are being performed with the application of the smaller length wavelet and not its whole video information is processed but only a part of it that is situated inside the contour of the previous hierarchical level.

For image row convolution it was proposed to use Haar's wavelet which executes multiplication of "+1" and "-1", which considerably simplifies the calculation and this way increases the efficiency.

When choosing the length of the wavelet, geometric sizes of the objects as constituent parts of the video information were taken into consideration. At the first stage the length of the wavelet had to correspond to the geometric size of the "biggest object" and when moving to the next stages it had to decrease its size according to the sizes of the object. A criterion to stop detailing is the reach of such level of detailing when the objects we are interested in would be selected.

As an advantage of the developed method are closed contours and also the possibility to ignore excessive video information with the help of video information convolution by the wavelet function. This allows to divide video information into a set of hierarchical detailing contours ("contour – sub contour"). And experimental studies showed high obstacle resistance of the proposed method according to the Pret's criterion (1,2 – 1,5 times higher when compared to the Kanny's method).

The method and basic informational technology of the contour inflection point selection with a regulated objects detailing were developed and in the basis of them rests the formation of the curvature function, after that the wavelet transformations of the curvature function is performed in the Haar's basis while as the inflection points are recognized the points that are situated on the intersection of the function's graph with the x axis. If the detailing level does not satisfy then the length of the wavelet is shortened and the described above procedure is repeated. Experimental studies showed that the developed method and basic informational technology substantially in 1,2 – 2 times surpass differential method according to obstacle resistance, and the interpolation method according to the accuracy – in 1,5 – 3 times.

In the third chapter, the method and basic informational technology of the structural and statistic object identification were improved on the basis of contour inflection points of the object with a regulated level of detailing. This method provides invariance to geometric transformations of the objects (dislocation in the field of view, rotation of the object and zooming) and possesses high obstacle resistance. As a result of the video information contour analysis on the basis of proposed approach pyramid presentation of the object's geometrical shapes and its details was obtained which includes a set of contour items. This presentation is used to calculate geometric identification features of the object. Given approach of the structural and hierarchical contour segmentation allows to increase operational speed of the video surveillance system due to the decrease of the information amount necessary for the processing.

Structural image performance as a set of primitive geometric elements (in our case it was a set of contours) and their relation between each other, was built on the basis of geometrical approach including mutual location and relations between certain elements into form. That is, an image may be presented as a set of contours of different hierarchical levels.

The essence of the methods is in the use of the identification vectors hierarchical sequence that were calculated on the basis of information about the object contour inflection points with the regulated detailing.

After the image was presented as a set of contours and their relations between each other on different hierarchical levels the selection of the inflection points, the construction of the identification vector takes place on the basis of the information about the contour inflection points, the conduction of the classification on the appropriate level and directing the object to the appropriate class (first on the contour of the highest hierarchical level). After that, a switch to the lower hierarchical level takes place and recognition within the class, in which the object ended up, is performed. This procedure lasts until the final classification decision is made.

Information technology of the structural and statistic identification of the hierarchical objects according to the contour IPs was created in the fourth chapter on the basis of the developed methods and basic contour IPs segmentation based on the Kanny's method of contour IPs selection with regulated detailing, which were used in video surveillance computer system.

Identification vector is calculated on the basis of geometrical moments-features method using the information exclusively concerning contour IPs. The following is attributed to the benefits of this method: invariance to parallel transfer, resistance to scale change, invariance to the object rotation angle, high stability of the method, signs are easily increased to obtain steady recognition and a small number of IPs determines a high computing efficiency of the method.

Objects identification according to the face is conducted on several levels: on the first level the object is referred to a certain class according to the shape of the head. If the given data is not sufficient, then the move to the next level of object hierarchy is performed (identification vectors of the nose, eyes and mouth contours).

On the basis of the proposed in chapters 2 and 3 methods within the scope of scientific and research work № 0112U008453 the AVSS was developed, whose software was implemented in Visual Studio 2015 environment on the basis of C# programming language using OpenCV library.

Developed ITs were installed in AVSS in private enterprise “Ecoderi”, Ltd. “VIP Computers”, Ltd. factory “REMPODUTTECHNIKA”, and also used in study process.

Studies showed that the implementation of these procedures in video surveillance information technology allowed to reduce the frame processing time in 8–15 times (depending on the frame content and the amount of objects in it) with a slight decrease in reliability (5-10%).

Keywords: automated video surveillance system, characteristic contour points, Haar wavelet, structural and statistic identification, identification vector.

List of the candidate’s publications

Articles in scientific and professional publications and foreign magazines:

1. Zahorodnia D. Investigation of the differential method in face contour inflection points selection / Zahorodnia D., Dorosh V., Dobrovolska N., Rymar O. // International scientific and technical magazine “Measuring and calculating equipment in technological processes.”– Khmelnytsk, 2014. – №4 (49). – P. 162-166.

2. Zahorodnia D. Face identification method taking into consideration the contour inflection points / Zahorodnia D. // Almanac of the Brest state technical university. Series: physics, mathematics, informatics. – 2015. – №5. – P. 30-33.

3. Zahorodnia D. Analysis of the interpolation method of the contour inflection points selection / Zahorodnia D. // Scientific Herald of the Chernivtsi

National University by Yuriy Fedkovych. Series: Computer systems and components. –2015. – Vol. 6, 2nd Edition. – P. 88-92.

4. Zahorodnia D. Comparison of the contour inflection points selection / Zahorodnia D. // Radio electronic and computer systems. Kharkiv "XAI". – 2016. – № 3 (77). – P. 56-61.

5. Zahorodnia D. Canny-based Method of Image Contour Segmentation / Diana Zahorodnia, Yuriy Pigovskyy, Pavlo Bykovyy // International Journal of Computing. Ternopil: TNEU. – 2016. – Vol. 15. – Issue 3. – P. 200-205. (*included into Scopus scientific metric base*).

6. Zahorodnia D. A method to recognize image contours using wavelet-transformations / Diana Zahorodnia, Uwe Großmann, Viktor Krylov, Vitaliy. Dorosh // Scientific Herald of the Chernivtsi National University by Yuriy Fedkovych. Series: Computer systems and components. – 2016. – Vol. 7, 2nd Edition. – P. 136-141.

Published works of the approbatory character:

7. Paliy Ihor. Fast and Robust Face Detection and Tracking Framework / Ihor Paliy, Volodymyr Dovgan, Ognian Boumbarov, Stanislav Panev, Anatoly Sachenko, Yuriy Kurylyak, Diana Zagorodnya // Proceedings of the IEEE 6th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS'2011). – Prague (Czech Republic), September 15-17, 2011. – Vol.1. – P. 430-433. (*included into Scopus scientific metric base*).

8. Zahorodnia Diana. Structural Statistic Method Identifying Facial Images by Countour Characteristic Points / Diana Zahorodnia, Yuriy Pigovsky, Pavlo Bykovyy, Viktor Krylov, Ihor Paliy, Igor Dobrotvor // Proceedings of the 8th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS'2015). – Warsaw (Poland), September 24-26, 2015. – P. 293-297. (*included into Scopus and Web of Science scientific metric bases*).

9. Bazelyuk D. Approach to emotion recognition on the basis of the face image. / Bazelyuk D., Zahorodnia D., Paliy I. // Works from the 13th international scientific and applied conference “Modern informational and electronic technologies” (“CIET-2012”). – Odesa, 4-8 June 2012. – P. 118.

10. Zahorodnia D. Contour tracking and inflection point selection algorithms in segmentation process of the computer recognition system / Zahorodnia D., Paliy I., Krylov V. // Modern computer information technologies: Materials from the 3rd Ukrainian school – seminar for young scientists and students (ACIT'2013). – Ternopil: TNEU, 17-18 May 2013. – P.129-130.

11. Zahorodnia D. Face recognition method using contour inflection points in video surveillance computer systems. / Zahorodnia D., Paliy I., Krylov V. // Works from the 14th international scientific and applied conference “Modern informational and electronic technologies” («CIET-2013»), 27-31 May 2013. – Odesa. – P. 49-52.

12. Zahorodnia D. Contour segmentation algorythms in video surveillance systems / Zahorodnia D., Kovalyuk K., Sachenko A., Krylov V. // Ukrainian scientific Internet-conference “Economical cybernetics – engineering of the economy (24th Edition)» / Collection of the abstracts from the reports: 24th Edition. – Ternopil: 29-30 November 2013. – P. 3-5.

13. Zahorodnia D. Contour Segmentation Method in Video Surveillance Systems / Diana Zahorodnia, Kostantin Kovalok, Anatoly Sachenko, Viktor Krylov, Sergiy Nychyporuk // Proceedings of the International Conference “Modern Problems of Radio Engineering, Telecommunications and Computer Science” (TCSET’2014). – Lviv-Slavske (Ukraine), February 25 – March 1, 2014. – P. 405.

14. Zahorodnia D. Configuration of the inflection points selection algorithm parameters on the basis of Haar’s wavelet / Zagorodnya D. // Modern computer and information technologies: Materials of the 5th Ukrainian school – seminar for young scientists and students ACIT'2015. – Ternopil: TNEU, 22-23 May 2015. – P. 68-70.

15. Zahorodnia D. Image inflection points selection methods on the basis of the curvature function wavelet-analysis / Zagorodnya D., Kovalyuk K., Krylov V. // Works from the 16th international scientific and applied conference “Modern

information and electronic technologies" ("CIET-2015"), 25-29 May 2015. – Odesa. – P. 50-51.

16. Zahorodnia D. Approach to the structural and hierarchical contour segmentation / Zahorodnia D. // Informatics, management and artificial intelligence. Materials of the 2nd scientific and technical conference students, Master's degree and postgraduate students. – Kharkiv: STU "KPI", 26-27 November 2015. – P. 31.

17. Zahorodnia D. Structural-Hierarchical Principle of Contour Segmentation / D.I. Zahorodnia, P.E. Bykovyy // Current Issues in Modern Technologies: Book of abstracts of the IV International Scientific and Technical Conference of Young Researchers and Students. – Ternopil: TNTU, 25-26 November 2015. – Vol. 2. – P. 20-21.

18. Zahorodnia D. Computer access control on the basis of the user's face image recognition / Zahorodnia D., Gubchakevych O., Kysil V. // Current tasks of the modern technologies: a collection of the abstracts from the reports of the Fifth international scientific and technical conference for young scientists and students,– Ternopil: TNEU, 17–18 November 2016. – Vol. 2. – P. 42-43.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

АСВН	автоматизовані системи відеонагляду
БД	база даних
ВД	відеодатчик
ГМО	геометричні моменти-ознаки
ІТ	інформаційна технологія
КН	кадровий накопичувач
КС	контурна сегментація
ХТ	характерна точка
CCTV	close-circuit television
DDL	data definition language
SDK	software development kit

ВСТУП

Актуальність теми. Стрімкий розвиток функціональних можливостей засобів обробки відеоінформації та відносно недорога їхня вартість привела до активного використання цифрових технологій у різних сферах людської діяльності. Зокрема, широкого розповсюдження набули системи комп’ютерного відеонагляду для спостереження в банках, офісах, аеропортах, супермаркетах, для пошуку суб’єктів в потоці людей за зовнішнім виглядом. В останні роки також спостерігається активне встановлення та використання відеокамер на зупинках громадського транспорту, в парках, скверах, площах, школах, прибудинковій території, тощо. Такі системи все частіше застосовують у криміналістиці, системах контролю доступу, охоронних системах.

В автоматизованих системах відеонагляду (АСВН) кількість камер постійно зростає, і відповідно ресурсоємність також збільшується. Проте суттєвим обмеженням таких систем являється необхідність великої кількості операторів для їх обслуговування. Це зумовило перехід до інтелектуальних АСВН. Проте, такі системи потребують великого об’єму обчислювальних затрат і значно знижують оперативність АСВН.

В таких системах, за допомогою прикладних інтелектуальних інформаційних технологій (ІТ), виділяються певні (підозрілі) об’єкти і ситуації. Враховуючи специфіку таких систем, іноді інформації про проблемні ситуації не достатньо для ухвалення рішення, тому необхідним є використання методів прийняття рішень за умов невизначеності, на які потрібно буде реагувати людині-оператору.

Для того, щоб підвищити оперативність АСВН потрібно зменшувати об’єм даних, що обробляються. Ця задача вирішується з допомогою інформаційної технології на етапах сегментації та ідентифікації об’єктів розпізнавання. Для вирішення цього завдання в роботі запропоновано використання методів вейвлет-аналізу, що дозволяють виділяти на відеоінформації об’єкти і деталі об’єктів різних геометричних розмірів, відповідно до заданої цілі обробки і, відповідно, використовувати ієрархічні методи та базові ІТ контурної сегментації, базові ІТ виділення характерних точок (ХТ) та базові ІТ ідентифікації, що в свою чергу дозволяє підвищити оперативність прикладних інтелектуальних інформаційних систем відеонагляду при достатньому рівні достовірності розпізнавання.

Основні наукові результати та висновки представлені в даній роботі були отримані на основі подальшого розвитку доробку багатьох вітчизняних (С.Г. Антощук [1-5], В.М. Крилов [1, 6-10], Г.Ю. Щербакова [6], М.В. Полякова [8-10], Є.П. Путятін [11-12], В.П. Машталір [13], О.М. Березький [14-16], Б.П. Русин [17-18], Д.Д. Пелешко [19-20], М.І. Шлезінгер [21], Й.Й. Білинський [22-23] та ін.) і зарубіжних вчених (С. Шімід [24], Ю. Тарабалка [25], У.Претт [26], Дж. Ліанг [27-28], Р. Вудс, Р. Гонсалес [29], Я.Фурман [30] та ін.).

Зв'язок роботи з науковими програмами, планами, темами. Основний зміст дисертаційної роботи складають теоретичні та практичні результати досліджень, проведені автором на кафедрі інформаційно-обчислювальних систем і управління та у Науково-дослідному інституті інтелектуальних комп'ютерних систем Тернопільського національного економічного університету при виконанні: науково-дослідної роботи «Виділення контурів для структурно-статистичної ідентифікації об'єктів» (№ державної реєстрації 0112U008453, відповідальний виконавець, 2013-2017 рр.), спільного Українсько-Молдовського науково-дослідного проекту «Дистрибутивні сенсорні мережі з реконфігурацією обчислювальних вузлів» (№ державної реєстрації 0115U004816), держбюджетного прикладного дослідження на тему «Методи захисту від комп'ютерних атак на основі нейронних мереж і штучних імунних систем» (№ державної реєстрації 0116U002499).

Мета і завдання дослідження. Метою дисертаційного дослідження є підвищення швидкодії методів та базових ІТ контурної сегментації і виділення характерних точок (ХТ) для ідентифікації облич в прикладних інтелектуальних ІТ комп'ютерного розпізнавання в умовах обмежених обчислювальних ресурсів.

Для досягнення даної мети необхідно вирішити наступні завдання:

– проаналізувати існуючі системи відеонагляду, методи виділення контурів, характерних точок та методів ідентифікації;

– розробити та експериментально дослідити метод та базову ІТ контурної сегментації;

– розробити та експериментально дослідити метод та базову ІТ виділення ХТ на основі властивостей вейвлет-аналізу з відповідним рівнем деталізації;

– розробити та експериментально дослідити метод та базову ІТ ідентифікації на основі ієрархічного принципу;

– розробити базові ІТ контурної сегментації, виділення ХТ та ідентифікації ієрархічних об’єктів, як складову прикладної ІТ для їх використання в АСВН, і провести аналіз їх достовірності та швидкодії.

Об’єкт дослідження – процес обробки візуальної інформації і прийняття рішень в умовах невизначеності у автоматизованих системах відеонагляду.

Предмет дослідження – методи та базові інформаційні технології виділення контурів, характерних точок контурів та ідентифікації ієрархічних об’єктів, зокрема облич людей отриманих з відеоінформації.

Методи дослідження – методи системного аналізу при розробленні узагальненої інформаційної моделі ідентифікації ієрархічних об’єктів; методи виділення контурів для підвищення достовірності і швидкодії процесу ідентифікації облич; вейвлет-аналіз, функціональний аналіз під час розробки методу виділення контурів та характерних точок контурів, методи математичної статистики при дослідженні достовірності запропонованих методів; методи моделювання та програмування при розробленні інформаційної технології ідентифікації облич.

Наукова новизна отриманих результатів. У ході виконання дисертаційного дослідження були отримані наступні основні результати:

– *вперше розроблено* метод та базову ІТ виділення ХТ контуру з регульованою деталізацією об’єктів, в основі яких лежить вейвлет-аналіз функції кривизни, що дозволило скоротити кількість точок контуру у відповідності з метою обробки;

– *отримав подальший розвиток* метод та базова ІТ контурної сегментації на базі методу Канні, відмінною особливістю яких є використання вейвлет-функцій для підкреслення перепадів інтенсивності зображення, що дозволило отримати послідовність контурних препаратів ієрархічного об’єкта з деталізацією, яка регулюється;

– *отримав подальший розвиток* метод та базова ІТ структурно-статистичної ідентифікації об’єктів на базі застосування ієрархічної послідовності ідентифікаційних векторів, розрахованих на основі характерних точок контуру об’єкта з регульованою деталізацією, що дозволило підвищити оперативність АСВН при високій достовірності;

– *отримала подальший розвиток* інформаційна технологія прийняття рішень про форму об’єктів ієрархічної структури на базі методів сегментації та ідентифікації з використанням вейвлет-аналізу, що дозволило підвищити оперативність АСВН.

Практичне значення отриманих результатів. Запропоновані у дисертаційній роботі методи реалізовано у прикладній ІТ. Вона дозволяє виявляти об’єкти у потоках відеоінформації з підвищеною оперативністю і достатньою точністю в умовах обмежених обчислювальних ресурсів. Ці результати досягнуто за рахунок використання методів та базових ІТ контурної сегментації, методів та базових ІТ виділення ХТ, методів та базових ІТ ідентифікації облич в прикладних інтелектуальних ІТ комп’ютерного розпізнавання на основі обмеженого числа точок перегину контуру (ХТ) і переходу до ієрархічного принципу ідентифікації та класифікації об’єктів, що використовуються для розпізнавання об’єктів з різним рівнем деталізації для підвищення швидкодії в умовах обмежених обчислювальних ресурсів.

Теоретичні та практичні результати роботи використані:

– при розробці програмного модуля контурної сегментації для АСВН на базі методу Канні (акт впровадження ПП «ЕКОДЕРІ» від 25.11.2016 р.);

– в програмному модулі АСВН, що реалізує метод виділення характерних точок контуру (акт впровадження ТОВ “ВІП комп’ютер” від 23.11.2016 р.);

– в системі відеонагляду, окремим програмним модулем структурно-статистичної ідентифікації об’єктів (акт впровадження ТзОВ завод «РЕМПОБУТТЕХНІКА», від 21.11.2016 р.);

– у науково-дослідній роботі на тему «Виділення контурів для структурно-статистичної ідентифікації об’єктів», державний реєстраційний № 0112U008453 (акт впровадження від 07.10.2016р.);

– у навчальному процесі при викладанні дисциплін “Комп’ютерна графіка”, “Теорія алгоритмів” та “Людино-комп’ютерна взаємодія” на кафедрі інформаційно-обчислювальних систем і управління, курсовому та дипломному проектуванні Тернопільського національного економічного університету (акт впровадження від 17.11.2016р.).

Особистий внесок здобувача. Усі основні результати, що виносяться на захист, отримані здобувачем особисто [31, 32, 33, 34, 35]. У роботах, опублікованих у співавторстві, здобувачу належать: [36] – реалізація диференціального методу виділення характерних точок контуру та проведення експериментальних досліджень його роботи; [37] – розробка методу контурної сегментації на основі методу Канні із застосуванням вейвлет-перетворення Хаара; [38] – проведення експериментальних досліджень методу виділення контурів зображень; [39] – проведення експериментальних досліджень для налаштування запропонованої платформи та комбінованого каскаду класифікаторів; [40] – реалізація структурно-статистичного методу ідентифікації облич для систем комп’ютерного розпізнавання; [41] – проведення огляду відомих рішень аналізу зображень облич; [42] – аналіз алгоритмів прослідковування контурів; [43] – проведення пошуку характерних точок контуру інтерполяційним методом; [44] – проведення аналізу процедури сегментації; [45] – проведення аналізу алгоритмів виділення та прослідковування контуру; [46] – розробка методу виділення характерних точок на основі вейвлет-аналізу функції кривизни; [47] – алгоритм структурно-ієрархічної контурної сегментації; [48] – узагальнена структура процесу розпізнавання облич.

Апробація результатів дисертації. Основні результати роботи були висвітлені та обговорені на: шостому та восьмому міжнародних симпозіумах “Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications” (IDAACS), Прага, Чеська республіка, 2011 р. та Варшава, Республіка Польща, 2015 р.; 13, 14 та 16 міжнародних науково-практичних конференціях “Сучасні інформаційні та електронні технології” (СІЕТ), Одеса, Україна, 2012 р., 2013 р. та 2015 р.; третій та п’ятій Всеукраїнській школі-семінарі молодих вчених і студентів «Сучасні комп’ютерні інформаційні технології» (ACIT), Тернопіль, Україна, 2013 р. та 2015 р.; Всеукраїнській науковій Інтернет-конференції «Економічна кібернетика – інженерія економіки», Тернопіль, Україна, 2013 р.; 12 Міжнародній науково-технічній конференції “Сучасні проблеми радіоелектроніки, телекомунікацій та комп’ютерної інженерії” (TCSET), Львів-Славсько, Україна, 2014 р.; другій науково-технічній конференції студентів, магістрів та аспірантів «Інформатика, управління та штучний інтелект», Харків, Україна, 2015 р.; IV та V міжнародній науково-технічній конференції молодих учених та студентів «Актуальні задачі сучасних технологій», Тернопіль, Україна, 2015 р. та 2016 р.

Публікації. Основні результати дисертаційної роботи знайшли повне відображення у 18 наукових роботах, які містять 5 статей у провідних фахових виданнях (з них 3 – у фахових виданнях зі списку спеціалізованих видань ВАК України, 1 стаття у закордонному фаховому виданні, 1 стаття у виданні, яке входить до міжнародної науко-метричної бази Scopus), 1 стаття у вітчизняному науковому журналі, 12 праць у збірниках міжнародних та всеукраїнських наукових конференцій, в тому числі 1 праця індексована науко-метричною базою Web of Science.

Структура дисертації. Дисертація складається зі вступу, чотирьох розділів, висновків, викладених на 116 сторінках основного машинописного тексту, списку використаних джерел (120 найменувань). Робота містить 36 рисунків, 8 таблиць та 8 додатків.

РОЗДІЛ 1
АНАЛІЗ ІСНУЮЧИХ СИСТЕМ ВІДЕОНАГЛЯДУ ТА МЕТОДІВ ІДЕНТИФІКАЦІЇ ІЄРАРХІЧНИХ ОБ'ЄКТІВ

1.1. Аналіз існуючих систем відеонагляду

Системи відеонагляду є одним з основних компонентів та займають важливе місце в загальній структурі комплексних систем безпеки об'єктів та фізичних осіб. У світі, де зростає рівень злочинності, бойових дій, нападів терористів та порушення безпеки, системи відеонагляду є відповідним рішенням для їх запобігання, виявлення та застереження [49].

На даний час, системи відеоспостереження все частіше інтегруються у різні аспекти повсякденного життя. Одним з варіантів класифікації таких систем є сфери їх використання [50]:

– відеонагляд за дорогами та автострадами: вимірювання швидкості автомобілів, виявлення проїзду на червоне світло світлофора, перетину розділювальної смуги та інших порушень правил дорожнього руху [51];

– громадська та комерційна безпека: моніторинг громадських місць для виявлення та попередження злочинності. Сюди включають як окремі об’єкти: школи, банки, супермаркети, театри, універмаги, автостоянки, стадіони так і цілі транспортні системи: аеропорти, залізниці, метро, морські порти та ін. [52];

– екологічний моніторинг та наукові дослідження: спостереження за лісовими пожежами та забрудненнями, місцями проживання та міграції тварин, гірськими хребтами, хворобами рослин, океанографічні дослідження, збереження історичних та археологічних пам'яток, культурної спадщини [53, 54];

– військова сфера: патрулювання державних кордонів, вимірювання потоків біженців, моніторинг мирного населення, забезпечення безпеки військових баз, допомога та управління під час бойових дій та ін.; [55]

– контроль якості: моніторинг промислових та автоматизованих процесів, виробничих майданчиків для виявлення несправностей та вторгнень в їхню інфраструктуру [56, 57];

– розумні будинки та особиста безпека: спостереження за будинком для запобігання крадіжок та вторгнень, станом здоров'я хворих, дітьми, тваринами та ін. [58-60];

– аналіз відеоінформації: визначення закономірностей та аномалій у русі транспорту, пішоходів, спортивних показників, рух торгових центрах, парках розваг та ін. [61]

З технологічної точки зору системи відеонагляду поділяються на чотири основні покоління [50]. Вони еволюціонували від систем керованих операторами, до систем автоматизованого та інтелектуального спостереження. Кожне покоління базується на попередньому проте додає певні відмінності як в апаратному так і програмному забезпеченні.

Системи раннього спостереження, або телевізійні системи, що призначені для обмеженого кола глядачів (Close-Circuit Television, CCTV), беруть всій початок з 70х років [49]. Вони складалися з однієї або кількох аналогових камер, підключених до відеомагнітофона (Video Cassette Recorder, VCR). У зв'язку з обмеженням часу запису на відеокасети, концепція режиму запису дозволяла відеомагнітофону записувати в кожну секунду: четверте, восьме або шістнадцяте зображення (генерація так званих кадрів на секунду, frames-per-second, fps). Цифрові відеореєстратори (Digital Video Recorders, DVRs) витіснили відеомагнітофони, замінивши відеокасети на жорсткі диски. Проте, незважаючи на покращення обладнання в таких системах відеоінформація, досі зберігалась як аналоговий сигнал.

Ера систем відеоспостереження другого покоління почалася з введення мережевих IP камер [62]. Такі системи є повністю цифровими, з високою якістю зображень та можливістю віддаленого управління: панорамування, нахил та масштабування. Для забезпечення постійного і якісного відеоспостереження, системи другого покоління [63, 64] використовували методи комп'ютерного зору для відображення інформації. Ці методи включають виявлення та відстеження об'єктів, аналіз подій, щоб допомогти оператору зосередити увагу лише на аномальних ситуаціях.

Третє покоління систем відеонагляду розпочалось з приходом дешевих високопродуктивних обчислювальних мереж та мобільного і мультимедійного зв'язку, де цифрова обробка інформації поширюється по всіх пристроях під'єднаних до мережі, відсутня єдина точка контролю та моніторингу яка замінена мережею розподілених камер і сенсорів. Інформація може бути зафіксована різнотипними сенсорами та різними способами (видимий, інфрачервоний, тепловий, аудіосигнал) [63, 65]. Найвища обчислювальна потужність досягається шляхом використання розподілених технологій, що дозволяє здійснювати обробку відео в режимі реального часу: виявлення, відстеження та аналізу поведінки об'єкта.

Четверте покоління характеризується переходом до вбудованих платформ з більш високою адаптивністю, ніж платформи на базі ПК. Тепер системи відеонагляду можна розділити на системи на базі ПК та на основі вбудованих платформ. Хоча діючі системи на базі ПК володіють великою ресурсоємністю [66], вони не можуть використовуватись для спостереження в «польових» умовах через великі розміри, енергоспоживання і малу надійність в несприятливих умовах [67, 68]. Починає розвиватись ринок спеціалізованих малопотужних акумуляторних вузлів [69, 70], здатних до відеоспостереження. Вони включають в себе можливості збору, обробки та передачі відеоінформації. Для цього вони оснащуються різного типу сенсорами (видимий, термічний, інфрачервоний, стереозвуковий), реконфігурованими модулями обробки (DSP, FPGA), акумуляторними модулями та модулями сонячних батарей, комунікаційними модулями, зокрема (WiMAX/4G, WiFi, 3G, Edge/GPRS [71, 72]).

Задачі, що виникають при побудові інтелектуальних систем відеонагляду, можна віднести до декількох основних областей. Перша з них пов'язана з поданням вихідних даних, отриманих як результати вимірювань для об'єкта, що підлягає розпізнаванню. Кожна виміряна величина є деякою характеристикою образу або об'єкта.

Вектори образів містять всю інформацію про образи, що піддається вимірюванню. Процес вимірювання, при якому об'єкти відносяться до певного класу образів, можна розглядати як процес кодування. Часто виявляється корисним розглядати вектори образів в якості точок *n*-мірного евклідового простору. Безліч образів, що належать одному класу, відповідають сукупності точок, розсіяних в деякій області простору вимірювань.

Друга задача - розпізнавання образів, пов'язана з виділенням характерних ознак або властивостей з отриманих даних і зниженням розмірності векторів образів. Це завдання часто визначають як завдання попередньої обробки зображення і вибору ознак.

Ознаки класу образів представляють собою характерні властивості, загальні для всіх образів даного класу. Ознаки, що характеризують відмінності між окремими класами, можна інтерпретувати як міжкласові ознаки. Міжкласові ознаки, загальні для всіх розглянутих класів, не несуть корисної інформації з точки зору розпізнавання і можуть не прийматися до уваги. Вибір ознак вважається однією з важливих задач підсистем розпізнавання. Якщо результати вимірювань дозволяють отримати повний набір розпізнавальних ознак для всіх класів то розпізнавання і класифікація образів не викличуть особливих труднощів. Автоматичне розпізнавання в такому випадку зводиться до процесу простого зіставлення або процедур типу перегляду таблиць. У більшості практичних задач розпізнавання як визначення повного набору розпізнавальних ознак є важкою задачею. Проте, з вихідних даних зазвичай вдається отримати певні дані з розпізнавальних ознак і використовувати їх для спрощення процесу автоматичного розпізнавання образів.

Третя задача - знаходження оптимальних процедур, необхідних для ідентифікації та класифікації. Після того як дані, зібрані про об’єкти, що підлягають розпізнаванню, представлені точками або ідентифікаційними векторами в просторі образів, потрібно системі з'ясувати до якого класу образів ці дані належать. Нехай система призначена для розпізнавання *M* класів, позначених w_1, w_2, ...w_m. В такому випадку простір образів можна вважати таким, що складається з *M* областей, кожна з яких містить точки, що відповідають образу з одного класу. При цьому задача розпізнавання може розглядатися як побудова границь областей рішень, які ділять на *M* класів, виходячи з зареєстрованих векторів вимірювань. Нехай ці межі визначені, наприклад, функціями $d_1(x)$, $d_2(x)$, … $d_M(x)$. Ці функції, звані також дискримінантними функціями, представляють собою скалярні і однозначні функції образу *x*. Якщо $d_i(x) > d_j(x)$ для всіх *i, j = 1, 2, ... M*, $i \neq j$, то образ *x* належить класу w_i . Іншими словами, якщо *i*-та вирішальна функція $d_i(x)$ має найбільше значення, то $x \in w_i$.

В загальному, система відеонагляду – це інформаційна система, яка складається з відеокамер, комплексу відображення і збереження відеоінформації, за допомогою яких здійснюється запис, перегляд та візуальний аналіз відеоінформації. Відеонагляд може проводитися як в режимі реального часу так і перегляду із збережених інших носіїв [49].

В свою чергу, інтелектуальна система відеонагляду – це система з власною операційною системою реального часу, яка забезпечує високу надійність в роботі і максимально ефективно використовує ресурси комп'ютера. Дозволяє добитися максимальної швидкодії, мінімального часу реакції на події і має довготривалу стабільність [49].

Такі системи характеризуються високою швидкістю роботи; великою кількістю камер; різними режимами відображення відеоінформації на екран; режимами контролю; автовізуалізація каналів; можливість масштабування; підтримка багатьох форматів запису; зміна параметрів зображення: яскравість, контрастність; інтелектуальний детектор руху; ведення журналу подій; робота за розкладом; каталог записаних кадрів.

Першим елементом інтелектуальної системи відеонагляду є відеодатчик. Прикладами відеодатчиків можуть бути цифрові або IP-камери. З відеодатчика зображення надходить на пристрій формування зображень. Після цього зображення проходить попередню обробку для покращення якості зображення і зменшення об’єму відеоінформації та потрапляє на кадровий накопичувач. Далі здійснюється локалізація області обличчя та проводиться контурна сегментація [42].

Контурна сегментація складається з: виділення контуру, прослідковування та виділення на ньому характерних точок. При виділенні контуру потрібно отримати зовнішній контур у вигляді замкнутої кривої, тому часто застосовують методи апроксимації [43]. Метод апроксимації кривих можна використовувати для контурної сегментації шляхом підбору аналітично заданої кривої до сукупності точок контурного препарату. Якщо є інформація про очікувану форму об'єкта, наприклад, як в нашому випадку (для ідентифікації облич), що це еліпс, то в якості апроксимуючої кривої потрібно взяти еліпс. Також можна спробувати застосувати апроксимацію поліномами, використовуючи ітеративні методи підбору.

Прослідковування контуру полягає у занесенні координат точок контуру у двовимірний масив. А власне сам алгоритм прослідковування контуру нагадує поведінку «жука» [44]. Для двійкового зображення уявний «жук» починає свій шлях на білому полі і рухається у напрямку до області чорних елементів зображення. Після того, як «жук» перетне чорний елемент, він повертається ліворуч і переходить до наступного елементу. Якщо цей елемент чорний, «жук» знову повертається ліворуч, якщо ж елемент виявляється білим, то «жук» повертається направо. Ця процедура триває доти, доки «жук» не повернеться у вихідну точку (контур замкнеться). Координати точок переходу з чорного на біле або з білого на чорне будуть координатами точок контуру.

Після цього виділяються характерні точки контуру (точки на ділянках значної кривизни [7]). На основі інформації про характерні точки контуру будується ідентифікаційний вектор (процедура ідентифікації) на основі якого далі проводиться класифікація – процедура віднесення (присвоєння) досліджуваного об'єкта (зображення) до певного класу [17].

Для виконання своїх функцій система комп’ютерного розпізнавання облич використовує ряд апаратно-програмних засобів, серед яких основними є відеодатчик і комп’ютер. Інтелектуальні відеодатчики можуть бути запрограмовані на виконання таких високорівневих процедур як попередня обробка та сегментація зображення, частково або повністю замінюючи собою комп’ютер [12].

Комп’ютер у системі розпізнавання облич представляє собою комплекс апаратного (мікропроцесор, оперативна пам'ять, дискові накопичувачі та інші) та програмного забезпечення (програмна підсистема виявлення, розпізнавання облич, підсистема управління базою даних облич, вхідні та вихідні файли та інші) і виконує, як правило, високорівневі процедури сегментації та аналізу зображень. У системах розпізнавання також може бути задіяне спеціалізоване апаратне забезпечення: плати відеозахоплення, мережеве обладнання для розподілених систем розпізнавання та інші [16].

Більшість із пропонованих на ринку автоматизованих систем відеонагляду забезпечують підтримку розглянутих вище процедур за допомогою розроблених апаратних та програмних засобів. Серед існуючих систем та засобів підтримки функції розпізнавання облич можна виділити наступні: системи контролю доступу, набори засобів та бібліотек для розробки програм з підтримкою функцій розпізнавання облич та системи роботи із відеоінформацією. Нижче наведені приклади використання таких систем.

Компанія „TiSO Production” (Україна), що професійно займається турнікетами, системами блокування проїздів, протипожежними дверима та системами паркування пропонує Турнікет Sweeper-AL для організації контролю доступу в пропускних терміналах аеропортів [73]. Цей турнікет може видавати готове рішення за рахунок використання інтегрованих зчитувачів магнітних карт (QR-кодів, відбитків пальця, розпізнавання облич і т.д.), що дозволить скоротити витрати часу на монтаж пункту пропуску.

Компанія Aurora (Великобританія) створила продукт для біометричного контролю доступу та аутентифікації – FaceSentinel [74]. Він використовує інфрачервоний сенсор з відповідною камерою для досягнення високої швидкості (менше 1 секунди), точності та надійності розпізнавання облич. Для автентифікації користувачам потрібно представити свої перепустки чи інші документи та обличчя до камери для верифікації. Дана технологія розпізнавання облич також використовується у найбільшому міжнародному аеропорту Великої Британії Хітроу для підтвердження пропуску на посадку.

Автоматичне розпізнавання відповідності обличчя на відеопослідовності можна здійснити за допомогою функцій Mobile Vision API компанії Google [75], що також дозволяють виявляти, відстежувати, визначати орієнтири (очі, рот, ніс), орієнтацію обличчя, класифікувати чи посміється людина. Дані функції отримують результат виходячи з положення та руху обличчя у відеопослідовності.

Компанія Animetrics пропонує наступні продукти [76]: FaceR API – набір функцій API для виявлення облич людей на знімках; ForensicaGPS – перевід 2D знімків облич людей в 3D аватари для подальшого їх виявлення на інших зображеннях; FIMS – система централізованого управління та сховища даних для 3D розпізнавання облич; ID-Ready – перетворення зображень облич зроблених під кутом (не більше ніж ±45°) у фронтальні.

Компанія Neurotechnology пропонує комплексні системи біометричної ідентифікації (MegaMatcher ABIS) та набори засобів та бібліотек для їх розробки (MegaMatcher Software Development Kit, SDK). Вони містять засоби та бібліотеки роботи з: відбитками пальців – VeriFinger SDK; звуком – Voice SDK; сітківкою ока – Eye iris SDK; обличчям: VeriLook SDK та SentiVeillance SDK (рисунок 1.1 (а)) [77].

Продукти компанії Cognitec включають технології обробки зображень, комп'ютерного зору та технології розпізнавання образів [78]. Технологія FaceVACS дозволяє клієнтам у всьому світі розробляти нові програми для розпізнавання обличчя (рисунок 1.1 (б). Вона забезпечує чіткий та логічний набір функцій API для розпізнавання облич: C ++, Java, Microsoft. NET, BioAPI 2.0 Verification Engine (C-API) для легкої інтеграції в інші програмні засоби.

Компанія Cognitec також підтримує розробку портативних прикладних програм для розпізнавання облич. Правоохоронні ограни використовують такі пристрої для здійсненні фотографій підозрюваних та порівняння їх з існуючими базами даних зображень.

а) SentiVeillance

б) FaceVACS

Рисунок 1.1 – Приклад ідентифікації та класифікації системами обробки відеозображень

Алгоритми VeriLook забезпечують вдосконалену локалізацію, реєстрацію та узгодженя облич з використанням надійних алгоритмів обробки цифрових зображень, які базуються на глибинних нейронних мережах [77]. Основними функціями є: одночасне виявлення і обробка кількох осіб як у відеопотоці так і на зображенні; гендерна класифікація; виявлення живого обличчя; розпізнавання емоцій; розпізнавання за особливими атрибутами (очі, посмішка, рот, окуляри, темні окуляри, борода, вуса і т.д.), оцінка віку, визначення якості зображення особи, ідентифікація особи.

Компанія Acsys Biometrics пропонує пакети засобів SDK, що дозволяють розробникам впроваджувати модулі роботи з біометричними технологіями, зокрема з обличчями, у своїх власних програмних продуктах. SDK забезпечує відстеження, реєстрацію, перевірку, класифікацію, базу даних, засоби зв'язку та мультимедійні елементи керування. А також супроводжується демонстраційними додатками зі зразком реєстрації, перевірки та класифікації вихідного коду, написаного на Visual C ++ [78].

Бібліотека комп'ютерного зору з відкритим вихідним кодом OpenCV містить алгоритми комп'ютерного зору, обробки зображень та алгоритми загального призначення з відкритим кодом [79]. Реалізована на C/C ++, також розробляється для Python, Java, Ruby, Matlab, Lua та інших мов програмування. Для роботи із відео та зображеннями OpenCV містить такі основні модулі: opencv_imgproc – для обробка зображень (фільтрація, геометричні перетворення, перетворення кольорових просторів і т. д.); opencv_features2d – розпізнавання та опис плоских примітивів; opencv_video – аналіз руху та відстеження об'єктів (оптичний потік, шаблони руху, видалення фону); opencv_objdetect – виявлення об'єктів на зображенні (виявлення осіб за допомогою алгоритму Віола-Джонса, розпізнавання людей HOG і та ін.).

Компанія Google запропонувала безкоштовний сервіс для зберігання, редагування та обміну фото і відео – Google Photos [80]. Сервіс автоматично аналізує фото, виявляючи різні візуальні особливості та предмети. Користувачі можуть здійснювати пошук по зображеннях. Google Photos дозволяє також розпізнавати обличчя, групуючи подібні разом; географічні ознаки і предмети використовуючи різні форми машинного навчання, зокрема розпізнавання вмісту фотографій, а також функцій, що дозволяють автоматично створювати альбоми, анімувати подібні фотографії в швидкі відеоролики, зберігати пам'ять над поверхнею впродовж значних часів та покращувати якість фотографій і відео.

Біометрична автоматизована інформаційно-пошукова система для правоохоронних органів “Сова-видеопоток” компанії AsiaSoftware дозволяє документувати затриманих контролюючими органами осіб, створювати картку контролю з персональними даними, здійснювати пошук за наявними в базі даних зображеннями, ідентифікувати осіб по опису, отримувати статистичну інформацію про наявних осіб [81].

Таким чином, у даному підрозділі показане широке розповсюдження систем відеонагляду, проаналізовано ряд існуючих зарубіжних і вітчизняних систем комп’ютерного розпізнавання облич. Показано, що системи відеоспостереження зазвичай мають складну розподілену структуру, та оперують великими обсягами відеоінформації, тому виникає складна задача систематизації та автоматизації обробки усієї відеоінформації.

Одним із підходів для вирішення цієї проблеми є залучення більш потужних обчислювальних засобів. Проте даний підхід не завжди є ефективним, оскільки у даний час для інформаційних систем існує багато різних за характеристиками апаратних засобів, в яких пріоритетною вимогою є забезпечення низького енергоспоживання (безпровідні відеокамери, і т.д.), що робить неможливим збільшення потужності обчислювальних ресурсів. Тому, більш перспективним є розроблення методів і технології розпізнавання об’єктів в умовах обмежених обчислювальних ресурсів.

Для цього використовують інформаційні інтелектуальні технології. Проте, не завжди системи цього класу мають достатньо ресурсів для повної та оперативної обробки усієї відеоінформації. Тому шляхи підвищення оперативності таких систем за рахунок зменшення об’єму відеоінформації, що обробляється є актуальною задачею.

Проведений аналіз показав, що існує можливість підвищення швидкодії розглянутих систем шляхом зменшення кількості інформації, що обробляється.

Оскільки при розпізнаванні об’єктів найбільш інформативною частиною зображення є контур, то для підвищення оперативності роботи планується перейти від обробки всього зображення до розпізнавання об’єктів за характерними точками контурів. Для цього буде проведена контурна сегментація, прослідковування контуру та виділення характерних точок контуру для подальшої класифікації об’єктів [7].

1.2. Аналіз відомих методів виділення контурів

При розпізнаванні об’єктів найбільш інформативною частиною зображення є контур. Контур об’єкта – це частина об’єкта, яка містить велику кількість інформації про форму об’єкта та мало залежить від кольору та текстури зображення [23, 26].

За контуром можна проаналізувати форму об’єкта. В багатьох випадках інформації про форму об’єкта достатньо для організації автоматичних або автоматизованих систем. Крім того, перехід до розпізнавання об’єктів за їх контурами дозволяє на декілька порядків зменшити об’єм інформації, що обробляється, крім того, контури інваріантні до трансформацій яскравості [17].

Оскільки основна інформація про форму об’єкта міститься в контурі об’єкта, то виділення та опис контуру є важливою задачею аналізу зображень.

Після оцифрування кожен піксель однозначно відноситься або до фону, або до зображення. Існують різні види критеріїв прийняття рішення про приналежність кожного з пікселів до фону або контуру зображення.

Контурні пікселі відрізняються від внутрішніх пікселів зображення наявністю в якості однієї або декількох сусідніх фонових комірок. При цьому в якості сусідніх розглядаються тільки горизонтальні та вертикальні сусідні комірки, або тільки діагональні [26]. Контур бінарного зображення задається точками в центрах клітин що межують. Якщо змістити решітку на величину половини довжини сторони квадрата вправо і вгору, то контурні точки зображення опиняться у вузлах решітки. В результаті сприйняття контуру спрощується.

Результатом виділення контурів є контурний препарат – вторинне зображення того ж розміру, що й вихідне. У початковий момент всі точки цього зображення мають чорний колір, а в процесі виділення контурів пікселі, що відповідають виявленим граничним точкам зображення, зафарбовуються білим кольором [82-87].

Контуру на кольоровому зображенні відповідає перепад інтенсивності. Проте, це визначення виключає контури, пов'язані з різкими змінами кольору тону та інтенсивності в областях з постійною яскравістю.

Представлення (кодування) контуру – це етап отримання дискретного сигналу, що описує границі оцифрованого зображення [29, 30].

Вимоги до алгоритмів представлення контурів:

– зменшення обсягів використовуваної пам'яті, для зберігання;

– зменшення часу та складності подальшої обробки;

– отримання інформативних ознак об’єкта.

Біологічні системи зорового сприйняття, як показують дослідження, використовують головним чином контури для виділення об’єктів, а не розділяють об'єкти по яскравості. На практиці перепади не будуть різкими зважаючи на розмивання і обмеження, що вносяться відеореєструючою апаратурою. Інколи перепади яскравості уздовж границь краще прослідковуються у вигляді стрибків перших похідних яскравості, ніж по аналізу значень самої яскравості [83].

При вирішенні задачі виділення контурів намагаються знайти компроміс між числом помилкових контурів та числом і величиною розривів контуру. Відомо, що на результат операції прослідковування набагато менше впливають невеликі розриви. Їх легше усунути, ніж помилкові контури, в яких легко заплутатися [84]. Співвідношення між числом помилкових контурів та числом і величиною розривів визначається завадостійкістю методу виділення контурів.

Будь-яка область D площини комплексної змінної містить внутрішні точки і точки контуру (граничні точки). Перші з них мають ту властивість, що не тільки вони самі, а й їх деякий окіл цілком належить області D. Точки контуру не є внутрішніми, але в будь-якому малому околі таких точок знаходяться внутрішні точки області D і точки, які не належать області D, – зовнішні (фонові) точки. Область D має властивість зв'язності, яка полягає у тому, що будь-які її точки з'єднуються лінією, яка повністю знаходиться в середині D [29].

Лінія контуру Г називається опуклою, якщо прямолінійний відрізок, що з'єднує дві її будь-які точки, цілком складається з внутрішніх точок області D. Ділянка контуру буде увігнутою, якщо такий відрізок буде включати зовнішні (фонові) точки (рисунок 1.2).

Внутрішній елемент (піксель) бінарного оцифрованого зображення $\omega(m_1, m_2)$ має властивість чотирьох-зв'язності, тобто суміжні з ним елементи – верхній, нижній, лівий і правий, теж належать $\omega(m_1, m_2)$.

Для обробки контуру аналітичним шляхом потрібно провести його кодування, тобто поставити у відповідність кожному контурному елементу певне число. Послідовність таких чисел називається кодом контуру.

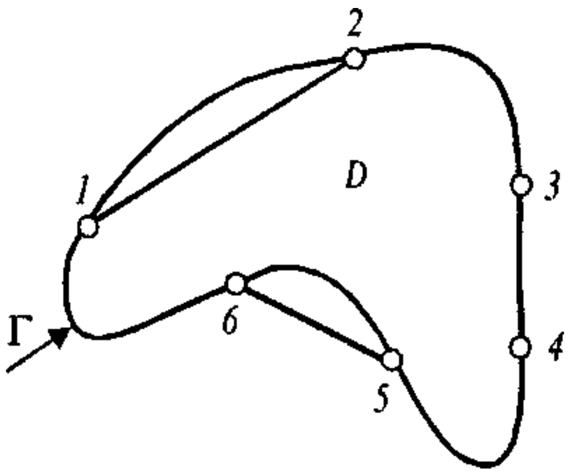


Рисунок 1.2 – Фрагменти контуру Г: 1, 2 – опуклий; 3, 4 – невизначений; 5, 6 – увігнутий

На квадратній решітці можливі вісім різних стандартних розміщень. Розглянемо деякі способи кодування контурів [30].

1. Кодування за трьома ознаками: довжині поточного елементарного вектора, напряму повороту при переході до наступного елементарного вектора і куту між сусідніми елементарними векторами.

2. Кодування поточного елементарного вектора трьохрозрядним двійковим кодом (числа від 0 до 7). Даний код був запропонований Фріменом і отримав широке поширення в задачах обробки зображень.

3. Кодування поточного елементарного вектора двома його проекціями на осі координат з початком відліку, поєднаним з початком елементарного вектора – двовимірний код .

4. Полігональне представлення контуру, що отримується при його апроксимації лінійними сегментами (рисунок 1.3).

Кодування полягає у фіксації координат кінців цих сегментів. Даний спосіб завдяки компактності одержуваних описів отримав широке поширення. При цьому виникає проблема сегментації, подібна з проблемою дискретизації сигналів. У реальних випадках вона зазвичай пов'язана з втратою інформації про форму зображень.

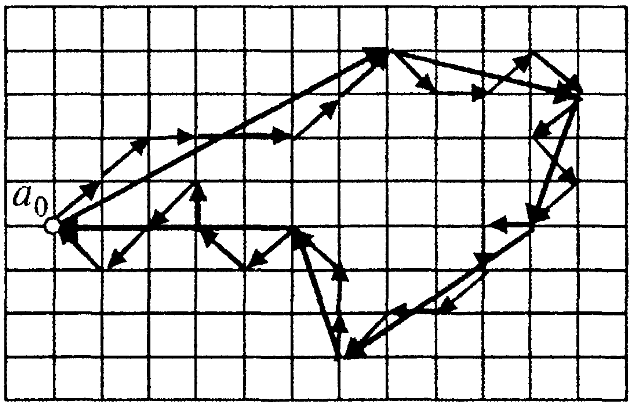


Рисунок 1.3 – Полігональне представлення контуру за допомогою апроксимації контуру лінійними сегментами

5. Подання лінії контуру полярним кодом [29]. На зображенні $\omega(m_1, m_2)$ вибирається полюс – початок відліку (т. Р) звичайної (власної) системи координат, тобто системи відліку, пов'язаної з цим зображенням. Центри всіх граничних точок зображення з'єднуються з т. Р. У результаті виходить

послідовність радіус-векторів $\beta(n)$, що однозначно задають контур зображення (рисунок 1.4). Часто центр суміщається з центром ваги зображення.

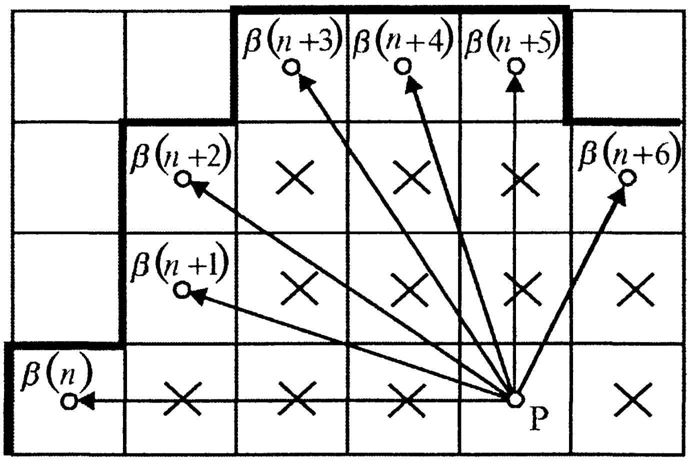


Рисунок 1.4 – Приклад задання фрагмента контуру в полярній систем
і координат

Методи виділення контурів можна розділити на дві великі групи: диференціальні та кореляційно-екстремальні [86]. В диференціальних методах перепади інтенсивності підсилюються за допомогою чисельного диференціювання, потім виділяється контур за допомогою порогового пристрою, після чого бінарне зображення піддається вторинній обробці, мета якої – потоншення контуру до одного пікселя. Методи прості в реалізації і мають високу швидкодію, але мають низьку завадостійкість. Головним критерієм при оцінці завадостійкості виділення контурів є положення перепаду яскравості.

З іншого боку, для визначення та опису контуру використовують два підходи: виділення країв або виділення області точки якої утворюють об’єкт.

У літературі наведено і описано велику кількість алгоритмів виділення контурів і границь. До найбільш популярних методів відносять: оператор Робертса, Собеля, Превітта, Кірша, Робінсона, алгоритм Канні і LoG-алгоритм [7, 11, 26, 29, 83-88]. Дані алгоритми базуються на підкресленні різких перепадів яскравості, які характерні контурам об’єктів.

Оператор Робертса [29] – один з перших алгоритмів виділення контурів, який обчислює суму квадратів різниць між діагонально-суміжними пікселями. Це буде згортка зображення з двома ядрами:

$$\begin{bmatrix} +1 & 0 \\ 0 & -1 \end{bmatrix} \quad \text{і} \quad \begin{bmatrix} 0 & +1 \\ -1 & 0 \end{bmatrix}. \tag{1.1}$$

Оператор Робертса все ще використовується заради швидкості обчислень, але він програє в порівнянні з альтернативами через значну чутливість до шуму. Він дає лінії тонші, ніж інші методи виділення контурів, що майже рівнозначно обчисленню кінцевих різниць уздовж координат X і Y. Іноді його називають «фільтром Робертса».

Дискретний оператор Лапласа [26] також часто використовується в обробці зображень для виділення контурів. Дискретний лапласіан визначається як сума других похідних і обчислюється як сума перепадів на «сусідах» центрального пікселя. Для одновимірного сигналу дискретний лапсасіан можна записати як згортку із наступним ядром:

$$D_x^2 = \begin{bmatrix} 1 & -2 & 1 \end{bmatrix} \tag{1.2}$$

А для двовимірного сигналу:

$$D_{xy}^2 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \text{ або } D_{xy}^2 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix} \tag{1.3}$$

Оператор Превіта [30] – метод виділення контурів в обробці зображень, який обчислює максимальне відхилення на безлічі ядер згортки для знаходження локальної орієнтації контуру в кожному пікселі. Він був створений доктором Джудіт Прюітт (Джудіт Превіт) для виявлення контурів медичних зображень.

Оператор використовує два ядра 3 × 3, згортаючи вихідне зображення для обчислення наближених значень похідних, одне по горизонталі, а друге – по вертикалі:

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -1 & 0 & +1 \\ -1 & 0 & +1 \end{bmatrix} \text{ і } G_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ +1 & +1 & +1 \end{bmatrix} \qquad (1.4)$$

Оператор Собеля [30] – дискретний диференціальний оператор, який обчислює наближене значення градієнта яскравості зображення. Використовується в області обробки зображень, зокрема, часто застосовується в алгоритмах виділення контурів. Результатом застосування оператора Собеля в кожній точці зображення є або вектор градієнта яскравості в цій точці, або його норма, що обчислюються за формулами:

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} * A \text{ і } G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix} * A \qquad (1.5)$$

де A – вхідне зображення.

Оператор Канні [82] – оператор виявлення контурів зображення. Був розроблений в 1986 році Джоном Кенні (Джон Канні), який використовує багатоступінчастий алгоритм для виявлення широкого спектра контурів на зображеннях. Канні вивчив математичну проблему отримання фільтра, за оптимальними критеріями виділення, локалізація і мінімізації. Він показав, що шуканий фільтр є сумою чотирьох експонент і може бути добре наближений першою похідною Гауссіана. Канні ввів поняття Non-Maximum, яке означає, що пікселями контурів оголошуються пікселі, в яких досягається локальний максимум градієнта в напрямку вектора градієнта. Хоча його робота була проведена на початках розвитку комп'ютерного зору, детектор контурів Канні досі є одним із кращих детекторів.

Результатом робити описаних алгоритмів є набір незв'язних областей. Для отримання зв'язного контуру необхідно провести додаткову обробку, наприклад морфологічну для отримання зв'язного краю об'єкта, який і називається контуром об'єкта.

Проведено порівняльний аналіз описаних методів виділення контуру облич. У порівняльному аналізі широко використовують оцінку якості зображень. Ця оцінка характеризується низкою метрик, які показують наскільки точно збігається отримане зображення із оригіналом. Найвідомішими із метрик є середньоквадратичне відхилення (MSE) та пікове відношення сигналу до шуму (PSNR) [23].

Середньоквадратичне відхилення (MSE) – це показник розсіювання значень випадкової величини відносно її математичного сподівання:

$$MSE = \frac{\sum_{i=1,j=1}^{m,n} \left(x_{i,j} - y_{i,j}\right)^2}{mn} \tag{1.6}$$

де MSE – середньоквадратичне відхилення; $x_{i,j}$ – елемент вибірки; $y_{i,j}$ – середнє арифметичне вибірки; *n, m* – об'єм вибірки.

Слід зазначити, що чим менше значення середньоквадратичного відхилення, тим більше отримане зображення збігається із оригінальним.

Пікове відношення сигналу до шуму (PSNR) – це відношення між максимумом можливого значення сигналу та потужністю шуму, що викривлює значення сигналу. Найпростіше визначити це відношення через середньоквадратичне відхилення:

$$PSNR = 10\log_{10}\left(\frac{MAX_I^2}{MSE}\right) \tag{1.7}$$

де MAX_I – максимальне значення інтенсивності пікселя по зображенню; MSE – середньоквадратичне відхилення.

Чим більше значення пікового відношення сигналу до шуму тим зображення вважається чіткішим та точнішим. Щоб оцінити якість отриманого зображення з виділеними контурами описаними методами та порівняти їх використано зображення із бази ORL [89]. Це зображення буде вважатись оригіналом, тобто в нашому випадку це ідеальне зображення. Оскільки досліджується робота детекторів виділення контуру, то це зображення було розмите та зашумлене за допомогою фільтра Гауса. В таблиці 1.1 наведені значення середньоквадратичного відхилення та пікового відношення сигналу до шуму для різних методів виділення контурів.

Таблиця 1.1 – Порівняльний аналіз роботи детекторів виділення контурів зображень

Критерії	Методи				
	Робертса	Лапласа	Превіта	Собеля	Канні
Середньоквадратичне відхилення	3508,14	11239,01	7297,56	11159,82	103,96
Пікове відношення сигнал/шум	12,68	7,62	9,5	7,65	42,16

Важливим показником роботи методів виділення контурів є швидкодія, яка вимірюється у секундах. У таблиці 1.2 наведені результати.

Таблиця 1.2 – Порівняння швидкодії відомих методів виділення контурів

Метод виділення контуру зображення	Швидкодія, мс
Робертса	0,60
Лапласа	0,86
Превіта	0,63
Собеля	0,76
Канні	2,45

Як видно із результатів, представлених в таблицях 1.1 і.1.2, метод Канні дає найкращі результати за середньоквадратичним відхиленням та піковим відношенням сигнал/шум, проте він і працює найдовше із розглянутих методів

Таким чином у підрозділі 1.2 показано, що основна інформація про об’єкт міститься в контурі, тому виділення та опис контуру є важливою задачею аналізу зображень, контури інваріантні до трансформацій яскравості, а перехід до розпізнавання об’єктів за їх контурами дозволяє на декілька порядків зменшити об’єм інформації, що обробляється.

Також було проведено порівняльний аналіз методів виділення контурів. Аналіз швидкодії, середньоквадратичного відхилення та пікового відношення сигналу до шуму показали, що метод Канні має найнижчу швидкість виділення контурів, проте за критеріями середньоквадратичного відхилення та пікового відношення сигналу до шуму показує найкращі результати.

1.3. Аналіз методів виділення характерних точок

Точки контуру представляють незначну частину всього зображення, проте і серед них існують такі, за допомогою яких можна ефективно описувати контур зображення об’єктів, оскільки при їх сполученні відрізками форма фігури зберігається [7].

Нехай заданий гладкий контур (він може бути отриманий, наприклад, з допомогою процедури прослідковування контуру). Аттніву, в 1954 р., провів психологічний експеримент, який полягав в тому, щоб розбити криву в точках найбільшої кривизни і сполучити точки перегину прямими лініями [26]. Цінність цього методу підтверджується відомим малюнком Аттніву (сплячий кіт), який побудований згідно описаного правила.

Такі точки, які передають суть фігури, знаходяться на ділянках значної кривизни контуру називаються ХТ контуру. Тому можна переходити до розпізнавання зображень використовуючи тільки інформацію про ХТ. Дослідження можливості ідентифікації зображень за допомогою набору ХТ почалися в минулому столітті з робіт по стереоприв'язці з використанням детектора кутів.

ХТ – це точки зображення, що володіють високою локальною інформативністю, тобто це точки максимуму, мінімуму, перегину і максимальної кривизни [7]. ХТ ще називають точками інтересу, ключовими точками, точками уваги, особливими точками, точковими особливостями, англійською мовою: salient, keypoints, representative, feature points, characteristic points, inflection points [90-94].

Прикладами ХТ будуть: кінці відрізка, вершини многокутників, точки перегину ламаної, точки перегину сплайнів, кінцеві точки півосей еліпса.

У 1992 Haralick і Shapir виділили наступні вимоги до ХТ [93]:

– відмінність (distinctness) – ХТ повинна явно виділятися на фоні і бути унікальною у своїй околиці;

– інваріантність (invariance) – визначення ХТ повинно бути незалежне від афінних перетворень;

– стабільність (stability) – визначення ХТ повинне бути стійким до шумів і помилок;

– унікальність (uniqueness) – крім локальної відмінності, ХТ повинна володіти глобальною унікальністю для поліпшення розрізнення повторюваних елементів;

– інтерпретованість (interpretability) – ХТ повинні визначатися так, щоб їх можна було використовувати для аналізу відповідностей і виявлення інтерпретованої інформації із зображення.

Tuytelaars і Mikolajczyk (2006) описали такі властивості, якими повинні володіти ХТ [94]:

– повторюваність (repeatability) – ХТ знаходиться в одному і тому ж місці сцени або об'єкта зображення, незважаючи на зміни точки огляду і освітленості;

– відмінність / інформативність (distinctiveness / informativeness) – повинен бути окіл ХТ, де вона повинні мати великі відмінності від решти точок в цьому околі, так, щоб можливо було виділити і зіставити особливі точки;

– локальність (locality) – ХТ повинна займати невелику область зображення, щоб зменшити ймовірність чутливості до геометричних і фотометрическим спотворень між двома зображеннями, знятих в різних точках огляду;

– кількість (quantity) – число виявлених ХТ має бути достатньо великим, так щоб їх вистачило для виявлення навіть невеликих об'єктів. Однак оптимальна кількість ХТ залежить від предметної області. В ідеалі кількість виявлених ХТ має адаптивно визначатися з використанням простого й інтуїтивного порога. Щільність розташування ХТ повинна відображати інформаційний вміст зображення, щоб забезпечити його компактне представлення;

– точність (accuracy) – виявлені ХТ повинні точно локалізувати, як у вихідному зображенні, так і взятому в іншому масштабі;

– ефективність (efficiency) – час виявлення ХТ на зображенні має бути допустимим в критичних за часом додатках.

В загальному, ці властивості перетинаються з попередніми, але по-різному інтерпретуються.

Виділення ХТ на зображенні відноситься до процедури КС. Основною перевагою використання ХТ для задач ідентифікації є відносна простота і швидкість їх виявлення.

Відомі наступні методи виділення ХТ:

– диференціальний;

– полігональний;

– інтерполяційний.

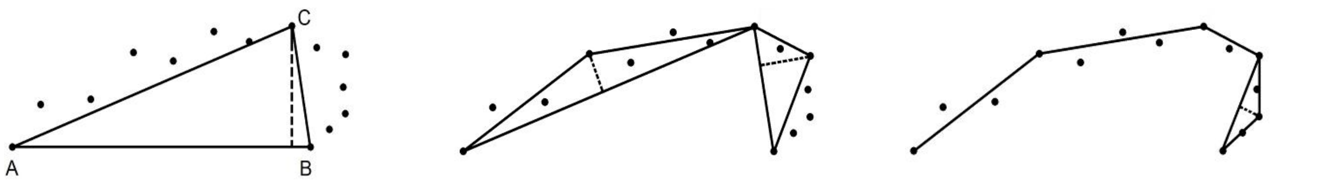
Диференціальний метод полягає в аналізі функції кривизни контуру, як диференціальної функція координат контуру, яка має властивості інваріантності до зсуву, повороту [7]. З математичної точки зору такий підхід пов'язаний з природнім рівнянням кривої, яке визначає її кривизну як функцію довжини дуги.

Отже, описаний метод еквівалентний встановленню природного рівняння кривої її екстремальними точками. ХТ є точки перегину та екстремуми функції кривизни та кінцеві точки. Для пошуку ХТ диференціальним методом потрібно побудувати функцію кривизни та дослідити її на екстремуми. Проте, в силу своєї диференціальної природи цей метод має низьку завадостійкість (не стійкий до шумів), а також не можна регулювати кількість виділених ХТ.

Полігональний метод полягає у послідовному вписуванні многокутників у контур зображення із наперед заданою точністю, а ХТ будуть вершини вписаного многокутника.

Реймер [95] використовував метод, схожий до процедури ітеративного підбору кінцевих точок для полігональної апроксимації замкнутих кривих довільної форми. Павлідіс і Хоровіц [96] розробили алгоритм для полігональної апроксимації кривих.

На першому етапі роботи алгоритму полігонального методу, представлені кінцеві точки A і B з'єднуються прямою лінією (рисунок 1.5). Потім шукається точка, відстань від якої до прямої що містить побудований відрізок є найбільшою (точка C). Якщо відстань велика (перевищує наперед задану точність), то ця точка відноситься до ХТ та береться в якості точки з'єднання двох відрізків (AC і CB). Дана процедура повторюється для кожного з відрізків до тих пір, поки всі точки контуру будуть знаходитись на відстані, яка менша за наперед задану точність.



а б в

Рисунок 1.5 – Ітеративний підбір кінцевих точок

а – перший етап; б – другий етап; в – третій етап

Проте алгоритм полігональної апроксимації складний в реалізації і не володіє достатньо великою швидкодією.

Інтерполяційний метод [7, 32, 43, 95] виділення ХТ контуру зображення полягає в підборі найдовшого прямолінійного відрізка між двома точками контуру так, щоб відстань від проміжних точок контуру до прямолінійного відрізка не перевищувала наперед заданого порогу DELTA.

Основні кроки алгоритму полягають в наступному (рисунок 1.6):

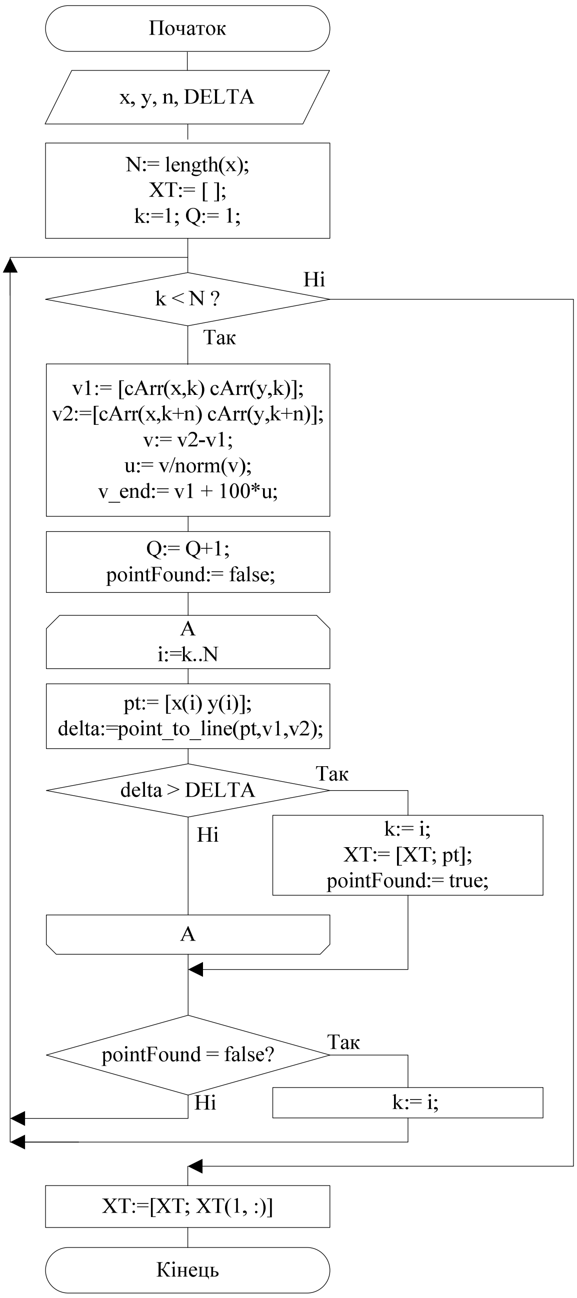


Рисунок 1.6 – Алгоритм виділення характерних точок контуру зображення на основі інтерполяційного методу

1. Вводиться масив координат прослідкованих точок контуру (*x* та *y*), довжина інтерполяційного відрізка (*n* – кількість точок) та поріг (*DELTA* – відстань від інтерполяційного відрізка до точки контуру).

2. Визначається кількість точок (*N* – довжина) вхідного масиву та оголошуються додаткові змінні.

3. Поки не пройдено усі точки контуру виконуються наступні операції:

3.1. Вибирається дві точки у вхідному масиві (*v1* та *v2*), що є початком та кінцем інтерполяційного відрізка довжиною в *n* точок. Через даний відрізок проводиться промінь з початком в точці *v1*.

3.1.1. В циклі послідовно (починаючи з наступної точки, що знаходиться після точки – початку інтерполяційного відрізка *v1*) обчислюється відстань (*delta*) від поточної точки контуру до інтерполяційного променя.

3.1.2. Якщо відстань від поточної точки до інтерполяційного відрізка (*delta*) перевищує заданий поріг (*DELTA*), то дана точка визначається як характерна, і заноситься в масив характерних точок (*XT*).

3.1.3. Здійснюється перехід до обробки наступних точок контуру (крок 3.1), під час якого знайдена характерна точка стає початком наступного інтерполяційного відрізка (*v1*).

4. Для графічного представлення, шляхом нанесення ХТ на контур зображення та їх послідовного сполучення, для замикання контуру перша характерна точка заноситься в кінець масиву та здійснюється вихід з алгоритму.

На рисунку 1.7 зображений чотирикутник (розмір вхідного зображення 92x76 пікселів): жирними точками позначено контур зображення (кількість точок – 82), інтерполяційні відрізки довжиною *n*=2 сполучені суцільною лінією, їх продовження зображено штриховою лінією. ХТ виділені даним алгоритмом додатково обведені (на представленому рисунку поріг *DELTA*=2).

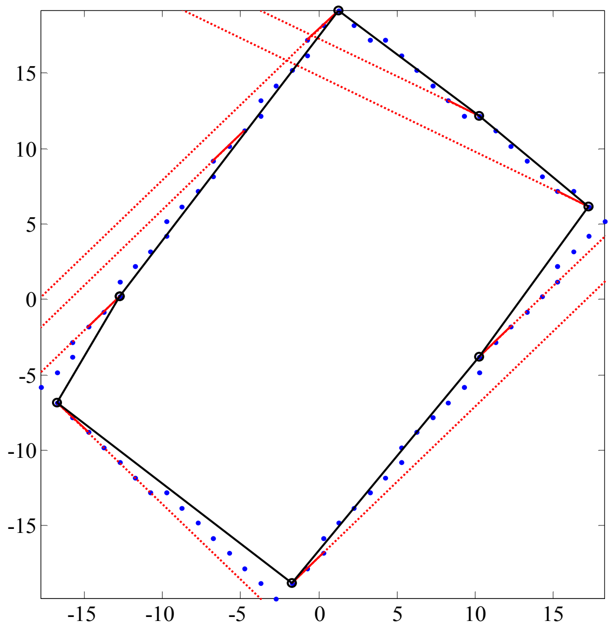


Рисунок 1.7 – Наочна демонстрація роботи інтерполяційного методу

Нехай $I = [(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)]$ – множина точок контуру зображення з центром $(0{,}0)$, впорядкована за їхнім кутом в полярних координатах $[\theta_1, \theta_2, \ldots, \theta_N]$, $\theta_i \le \theta_{i+1}$, $i = 1{,}2{,}...{,}N-1$; I_{EXT}, I_{INT} – підмножини точок контуру, що мають максимальні і мінімальні значення полярного радіусу для певного полярного кута:

$$I_{EXT} = \left[(x_i, y_i) \middle| i = \arg\max_i \rho_i, \theta_i = \theta, \forall\theta \in \Theta \right], \qquad (1.8)$$

$$I_{INT} = \left[(x_i, y_i) \middle| i = \arg\min_i \rho_i, \theta_i = \theta, \forall\theta \in \Theta \right], \qquad (1.9)$$

де Θ – підмножина унікальних значень полярних кутів $\Theta \subseteq [\theta_1, \theta_2, \ldots, \theta_N]$; $\rho_i \in [\rho_1, \rho_2, ..., \rho_N]$ – множина полярних радіусів точок I.

Введемо підмножину ХТ контуру Γ ($\Gamma \subseteq I$), тоді якість множини ХТ можна оцінити за формулами:

$$Q_1 = \int\limits_{-\pi}^{\pi} \left(F_{EXT}(\theta) - C(\theta)\right) h\left(F_{EXT}(\theta) - C(\theta)\right) d\theta, \tag{1.10}$$

$$Q_2 = \int\limits_{-\pi}^{\pi} \left(C(\theta) - F_{INT}(\theta)\right) h\left(C(\theta) - F_{INT}(\theta)\right) d\theta, \tag{1.11}$$

де $F_{EXT}(\theta)$ та $F_{INT}(\theta)$ – інтерполяційні функції, що апроксимують полярний радіус точок I_{EXT}, I_{INT} для аргументу $-\pi \le \theta \le \pi$; $C(\theta)$ – інтерполяційна функція, що апроксимує полярний радіус ХТ Γ для аргументу $-\pi \le \theta \le \pi$; $h(x)$ – функція Хевісайда:

$$h(x) = \begin{cases} 1, при\, x > 0, \\ 0, при\, x \le 0. \end{cases} \tag{1.12}$$

Інтерполяцію функцій $F_{EXT}(\theta)$, $F_{INT}(\theta)$ та $C(\theta)$ можна виконати таким чином:

якщо $\theta > \theta_N$ або $\theta < \theta_1$, то шукаємо точку (x', y') перетину відрізку $(x_1, y_1) - (x_N, y_N)$ з бігучою прямою інтегрування $y = x \tan\theta$, тоді

$$\rho = \sqrt{(x')^2 + (y')^2}, \tag{1.13}$$

де ρ – це точкове значення однієї з функцій $F_{EXT}(\theta)$, $F_{INT}(\theta)$ чи $C(\theta)$;

інакше, якщо $\theta_1 \le \theta \le \theta_N$, то знаходимо на множині полярних кутів $[\theta_1, \theta_2, \ldots, \theta_N]$ інтервал $\theta_i \le \theta \le \theta_{i+1}$, $i = 1,2,..., N-1$ і шукаємо точку (x', y') перетину відрізку $(x_i, y_i) - (x_{i+1}, y_{i+1})$ з бігучою прямою інтегрування $y = x \tan\theta$, причому $\tan\theta = tg(\theta + \pi)$, якщо $tg(x)$ – функція тангенсу для $0 \le x \le 2\pi$.

Рівняння прямої $(x_i, y_i) - (x_{i+1}, y_{i+1})$ записується так:

$$\begin{cases} y_i = ax_i + b, \\ y_{i+1} = ax_{i+1} + b, \end{cases} \tag{1.14}$$

де $(x_i, y_i), (x_{i+1}, y_{i+1})$ – відомі координати крайніх точок відрізку, a, b – невідомі коефіцієнти.

Віднімаємо від першого рівняння друге:

$$y_i - y_{i+1} = a(x_i - x_{i+1}), \tag{1.15}$$

Звідки

$$\frac{y_i - y_{i+1}}{x_i - x_{i+1}} = a, \tag{1.16}$$

$$y_i - \frac{y_i - y_{i+1}}{x_i - x_{i+1}} x_i = b, \tag{1.17}$$

Точка перетину (x', y') знаходиться з рівняння

$$ax' + b = x' \tan\theta, \tag{1.18}$$

$$b = x'(-a + \tan\theta), \tag{1.19}$$

$$\frac{b}{-a + \tan\theta} = x', \tag{1.20}$$

$$y' = a\frac{b}{-a + \tan\theta} + b. \tag{1.21}$$

У випадку, коли $x_i = x_{i+1}$, відрізок $(x_i, y_i) - (x_{i+1}, y_{i+1})$ є вертикальним, і тоді точка перетину обчислюється за такими формулами:

$$x' = x_i = x_{i+1}, \tag{1.22}$$

$$y' = x_i \tan\theta, \tag{1.23}$$

коли ж $\theta = \dfrac{\pi}{2}$ або $\theta = \dfrac{3\pi}{2}$, то бігуча пряма інтегрування є вертикальною і тоді точка перетину визначається так:

$$\begin{aligned} x' &= 0, \\ y' &= b. \end{aligned} \tag{1.24}$$

Поняття характерних точок та бігучої прямої інтегрування проілюстровано рисунком 1.8.

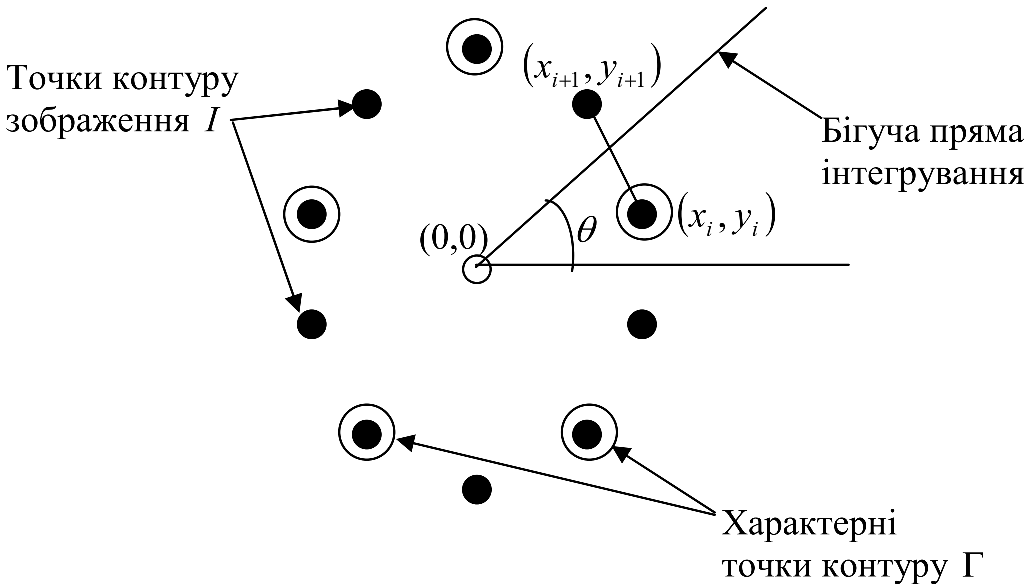


Рисунок 1.8 – Поняття ХТ та бігучої прямої

Загалом, виділення ХТ в даний час залишається предметом активних досліджень. Основним алгоритмічним апаратом, що застосовується для виділення ХТ, є аналіз функції кривизни контуру. В силу своєї диференціальної природи вона має низьку завадостійкість. До недоліків полігонального методу відносять складність в реалізації та високі вимоги до обчислювальних затрат, а в інтерполяційному методі – низька точність виділення ХТ.

Тому актуальною залишається задача розробки методу виділення ХТ, який володіє наступними властивостями:

- завадостійкий;
- простий в реалізації;
- висока швидкодія;
- точний у визначенні координат характерних точок.

Висновки до розділу 1

У результаті аналізу існуючих систем відеонагляду, областей їхнього застосування, відомих методів виділення контурів та методів виділення ХТ, отримано наступні висновки:

1. Застосування різнотипних засобів відеоспостереження стає все більш популярним в зв’язку із збільшеннями їхніх функціональних можливостей. Це дає змогу організовувати відповідну безпеку як приватної особи чи майна, так і безпеку особливо важливих об’єктів чи об’єктів підвищеної небезпеки.

2. Збільшення відеопотоків веде до потреби додаткового їх опрацювання, що, в свою чергу, вимагає додаткових затрат на обчислювальні чи людські ресурси. Тому, методи, що дозволяють зменшити об’єми оброблюваної інформації стають все більш актуальними.

3. У контурі зображення міститься інформація про його форму, що є важливим фактором для сприйняття та розпізнавання образів. Перехід до розпізнавання об’єктів за їх контурами дозволяє на декілька порядків зменшити об’єм інформації, що обробляється, внаслідок чого розпізнавання зображень буде проводитись набагато швидше.

4. Точки контуру представляють незначну частину всього зображення, проте і серед них існують такі ХТ, за допомогою яких можна ефективно

описувати контур зображення об’єктів, оскільки при їх сполученні відрізками форма фігури зберігається (теорема Аттніва). Тому доцільно використовувати методи виділення ХТ, що в свою чергу знову ж таки зменшить об’єм інформації, що обробляється.

5. На базі узагальненої інформаційної моделі ідентифікації ієрархічних об’єктів сформовано наступні завдання, які необхідно вирішити у дисертаційній роботі для підвищення швидкодії роботи АСВН в умовах обмежених обчислювальних ресурсів:

– розробити та експериментально дослідити метод та базову ІТ контурної сегментації;

– розробити та експериментально дослідити метод та базову ІТ виділення ХТ;

– розробити та експериментально дослідити метод та базову ІТ ідентифікації;

– розробити базові ІТ контурної сегментації, виділення ХТ та ідентифікації ієрархічних об’єктів, як складову прикладної ІТ для їх використання в АСВН, і провести аналіз їх достовірності та швидкодії.

Основні результати розділу опубліковані в працях [36, 32, 39, 41].

РОЗДІЛ 2
МЕТОДИ ВИДІЛЕННЯ КОНТУРІВ ЗОБРАЖЕНЬ ТА ЇХ ХАРАКТЕРНИХ ТОЧОК В ПРОСТОРІ ВЕЙВЛЕТ-ПЕРЕТВОРЕНЬ

2.1. Метод контурної сегментації

Контури зображень є областями з високою концентрацією інформації, яка мало залежить від кольору та інтенсивності. При розгляді будь-якого об'єкта в свідомості людини формується зоровий образ. При сприйнятті очима відстежується лінія контуру, що призводить до створення в свідомості образу. Контурний аналіз є сукупністю методів виділення, опису та перетворення контурів зображень та розпізнавання зорових образів [30].

Контур цілком визначає форму зображення і містить всю необхідну інформацію для розпізнавання зображень за їх формою. Такий підхід дозволяє не розглядати внутрішні точки зображення і тим самим значно скоротити обсяг інформації, що обробляється. Наслідком цього часто стає можливість забезпечення роботи системи в режимі реального часу.

Контур – просторово-протяжний розрив, перепад або стрибкоподібна зміна значень інтенсивності. Існує ряд проблем при виділенні контурів зображення: розриви контуру в місцях, де інтенсивність змінюється не дуже швидко; наявність помилкових контурів внаслідок шуму на зображенні; широкі контурні лінії через розмитість або шум [26].

Контурний аналіз дозволяє описувати, зберігати, порівнювати і робити пошук об'єктів, представлених у вигляді своїх зовнішніх контурів, а також ефективно вирішувати основні проблеми розпізнавання образів - перенесення, поворот і зміна масштабу зображення об'єкта [30].

Однією з головних цілей обробки зображень є інтерпретація вмісту на зображенні. Для цього необхідно відокремити фон від об'єктів. Сегментація дозволяє розділити зображення на складові частини або об'єкти, вона відокремлює об'єкт від фону, щоб можна було легко обробляти зображення і ідентифікувати його вміст. Виділення контурів на зображенні є фундаментальним засобом для якісної сегментації зображення. Вони в значній мірі зменшують кількість даних для обробки зображення, зберігаючи при цьому важливу інформацію про об'єкти на зображенні, таку як їх форму, розмір, кількість [96]. А ступінь деталізації, до якої проводиться такий поділ, залежить від мети поставленої задачі, тобто, сегментацію слід припинити, коли об'єкти, що цікавлять виявляються ізольованими.

Як правило, алгоритми сегментації зображень грунтуються на одному з двох базових властивостей яскравості сигналу: однорідності і перепадах інтенсивності [7].

Проте на даний час не існує універсального методу чи алгоритму виділення контурів зображень. Для визначення відповідного алгоритму виділення контурів враховують орієнтацію та структуру контуру, а також наявність і тип шуму на зображенні. Кожен з алгоритмів вирішує свій клас завдань, якісно виділяючи лише контур певного типу, тому задача створення методів контурної сегментації зображення є актуальною.

Проаналізувавши методи виділення контурів, в якості базового методу був обраний метод Канні [82], до переваг якого належить виділення контуру товщиною в 1 піксель. Проте, даний метод не підходить для реалізації ієрархічного підходу. Тому при реалізації цього підходу потрібна морфологічна обробка отриманого контурного препарату, що суттєво знижує оперативність. В даній роботі розроблений наступний ієрархічний метод контурної сегментації.

Найбільш загальним способом пошуку контурів зображення є обробка зображення за допомогою накладання маски. Проте замість маски Собеля, як зроблено в методі Канні, запропоновано використати вейвлети [97-101].

Ієрархічний метод КС полягає в реалізації наступних кроків [37-38]:

1) Згортка рядка зображення вейвлет-функцією.

Для вейвлету розміром n, ця процедура заснована на обчисленні лінійної комбінації зі значеннями інтенсивності пікселів зображення, що покриваються вейвлетом.

Для покращення збіжності вейвлет-перетворення, виконується доозначення крайнього ряду пікселів, шляхом їхнього копіювання на довжину вейвлета:

$$z_{i,j} = \begin{cases} I_{i,1}, \ \textit{якщо } j < n, \\ I_{i,M}, \textit{якщо } \ j > M + n \\ I_{i,j-n+1}, \textit{ в іншому випадку} \end{cases} \tag{2.1}$$

де $I_{i,j}$ - піксель оригінального зображення, $i = 1,2,...N$, $j = 1,2,...M$; $z_{i,j}$ - піксель доозначеного зображення, $i = 1,2,...N$, $j = 1,2,...M + 2n$; n - довжина вейвлета.

Для кожної точки зображення обчислюємо:

$$WX_{i,j} = \sum_{k=i-n}^{i-1} w_k z_{k,j} + \sum_{k=i+1}^{i+n} w_k z_{k,j} \tag{2.2}$$

де $z_{k,j}$ - значення яскравості (інтенсивності) пікселя, що відповідає коефіцієнту w_k вейвлету. Результат вейвлет-перетворення записується в центральний елемент.

Аналогічно доозначуємо зображення по осі OY:

$$z_{i,j} = \begin{cases} I_{1,j}, \ \textit{якщо } \ i < n, \\ I_{N,j}, \textit{якщо } i > N + n \\ I_{i-n+1,j}, \textit{ в іншому випадку} \end{cases}, \tag{2.3}$$

де $I_{i,j}$ - піксель оригінального зображення, $i = 1,2,...N$, $j = 1,2,...M$; $z_{i,j}$ - піксель доозначеного зображення, $i = 1,2,...N + 2n$, $j = 1,2,...M$; n - довжина вейвлета.

Такі ж обчислення, що й у формулі (2.2) проводимо і по осі OY:

$$WY_{i,j} = \sum_{k=j-n}^{j-1} w_k z_{i,k} + \sum_{k=j+1}^{j+n} w_k z_{i,k}, \tag{2.4}$$

В результаті отримаємо дві матриці того ж розміру, що й вхідне зображення: $WX_{i,j}$ та $WY_{i,j}$.

При згортці зображення вейвлет-функцією по осі *OX*, найбільші значення будуть на вертикальних лініях товщиною в 1 піксель. Причому, на областях зображення з постійною інтенсивністю значення буде дорівнювати 0, оскільки сума коефіцієнтів вейвлету дорівнює нулю. Аналогічно, при згортці стовпця зображення вейвлет-функцією по осі *OY* – найкраще будуть виділятись горизонтальні лінії.

2) Встановлення порогу (пороговий фільтр).

Поріг встановлюється для значення інтенсивності пікселів: якщо значення інтенсивності менші за поріг, то заміняємо її на 0, ту ж, що вища порогу – залишаємо без змін.

В результаті знову отримаємо дві матриці того ж розміру, що й вхідне зображення $WX'_{i,j}$ та $WY'_{i,j}$:

$$WX'_{i,j} = \begin{cases} WX_{i,j}, \ \textit{якщо} \ WX_{i,j} > P \\ 0, \ \textit{в іншому випадку} \end{cases} \tag{2.5}$$

$$WY'_{i,j} = \begin{cases} WY_{i,j}, \ \textit{якщо} \ WY_{i,j} > P \\ 0, \ \textit{в іншому випадку} \end{cases} \tag{2.6}$$

де P – значення порогу.

Після цього переходимо до пошуку перепадів інтенсивності. Перепад – це зв’язана тим чи іншим критерієм множина пікселів, що лежать на межі між двома областями.

3) Морфологічна обробка по градієнту.

Обчислюємо величину градієнта для визначення величини стрибка інтенсивності та його напрям. Контур відзначається там, де градієнт зображення набуває максимального значення.

Величина градієнта характеризує швидкість зміни функції *f* в точці (*x, y*) та обчислюється за формулою:

$$\begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \dfrac{\partial f}{\partial x} \\ \dfrac{\partial f}{\partial y} \end{bmatrix} \tag{2.7}$$

Напрям вектора градієнта співпадає з напрямом максимальної швидкості зміни функції в точці та обчислюється за формулою:

$$\nabla f = \sqrt{G_x^2 + G_y^2} \tag{2.8}$$

Ця величина дорівнює значенню максимальної швидкості зміни функції *f* в точці, причому максимум досягається в напрямку вектора ∇f. Величину ∇f часто називають градієнтом. Напрям вектора градієнта також є важливою характеристикою. Позначимо $\alpha(x, y)$ - кут між напрямком вектора ∇f в точці (*x, y*) і віссю *OX*, отже:

$$\alpha(x, y) = arctg\left(\frac{G_y}{G_x}\right) \tag{2.9}$$

Звідси легко знайти напрям контуру в точці (*x, y*), що перпендикулярний до напрямку вектора градієнта в цій точці.

Тільки максимуми величини градієнта позначаються як границі. До контуру відносимо пікселі, в яких досягається максимум градієнта в напрямку вектора градієнта.

Дані кроки проводимо окремо для двох матриць $WX'_{i,j}$ та $WY'_{i,j}$, а отриманий контурний препарат складаємо за схемою логічного «або» (рисунок 2.1).

Після того як отримали перший контур (найвищого рівня ієрархії), для отримання контуру наступного рівня ієрархії, зменшуємо довжину вейвлета та переходимо до кроку 1. Але вже працюємо не із всім зображенням, а тільки з тою частиною, що знаходиться в середині контуру вищого на один порядок рівня ієрархії.

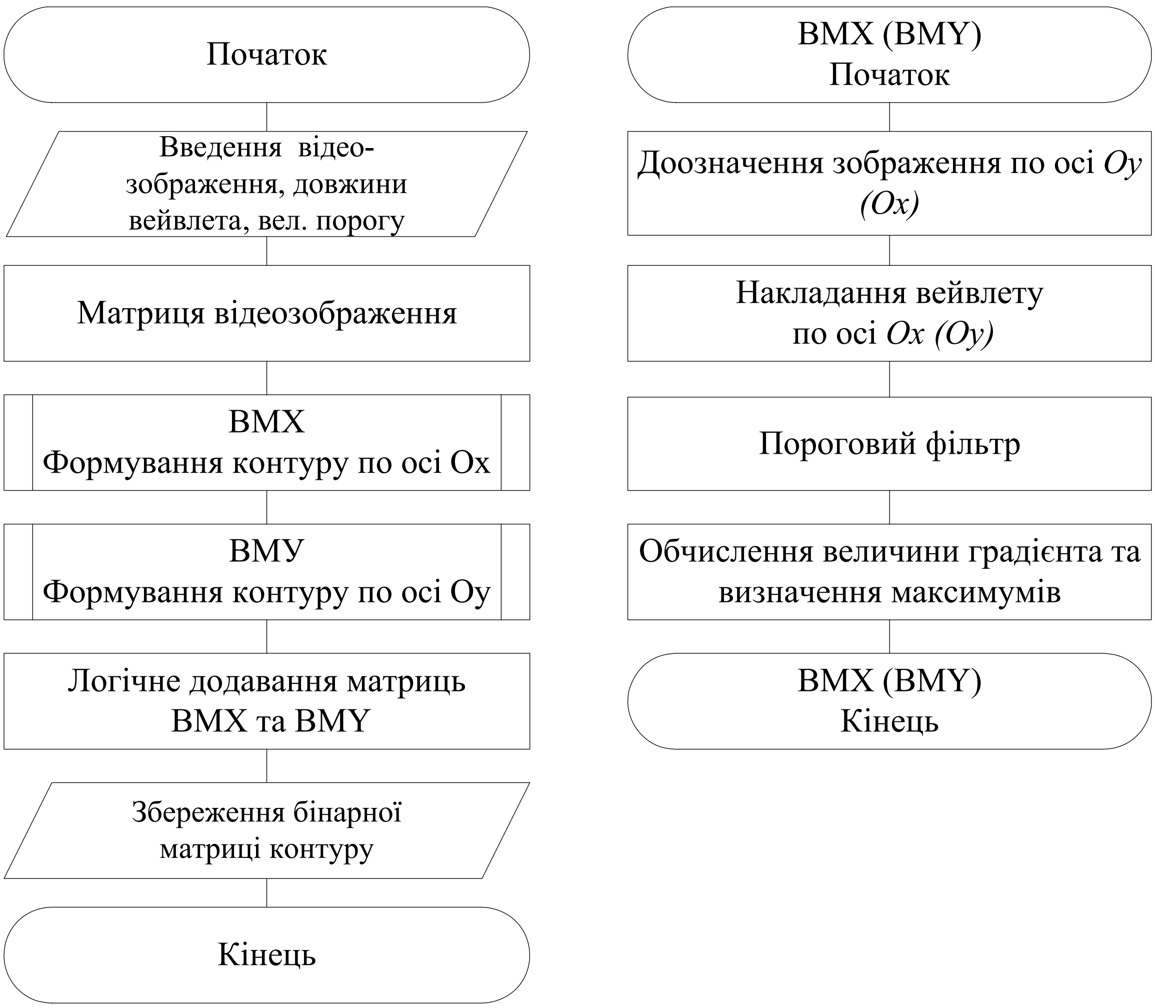


Рисунок 2.1 – Схема методу виділення контурів

При виборі довжини вейвлету потрібно керуватись геометричними розмірами об’єктів, з яких складається зображення. На першому етапі довжина вейвлету повинна відповідати геометричним розмірам «найбільшого об’єкта» і з переходом на наступні рівні повинна зменшуватись відповідно до розмірів об’єкта. Критерієм виходу з циклу буде досягнення потрібного рівня деталізації. Ступінь деталізації, до якої проводиться такий поділ, залежить від мети поставленої задачі, тобто, сегментацію слід припинити, коли об'єкти, що цікавлять виявляються виділеними.

Перевагою розробленого методу є замкнутість контурів, а також можливість ігнорування надлишковою відеоінформацією за допомогою згортки відеоінформації вейвлет-функцією. Це дозволяє розбити відеоінформацію на сукупність ієрархічних контурів деталізації («контур – підконтур»).

Запропонований метод реалізований в пакеті прикладних програм Matlab у вигляді виконуваних m-скриптів.

Перетворення зображення виконується за допомогою вейвлету Хаара (рисунок 2.2), оскільки він володіє низькою обчислювальною складністю, реалізуючи множення на +1 та -1, що значно спрощує його апаратну реалізацію та, відповідно, підвищує оперативність [97].

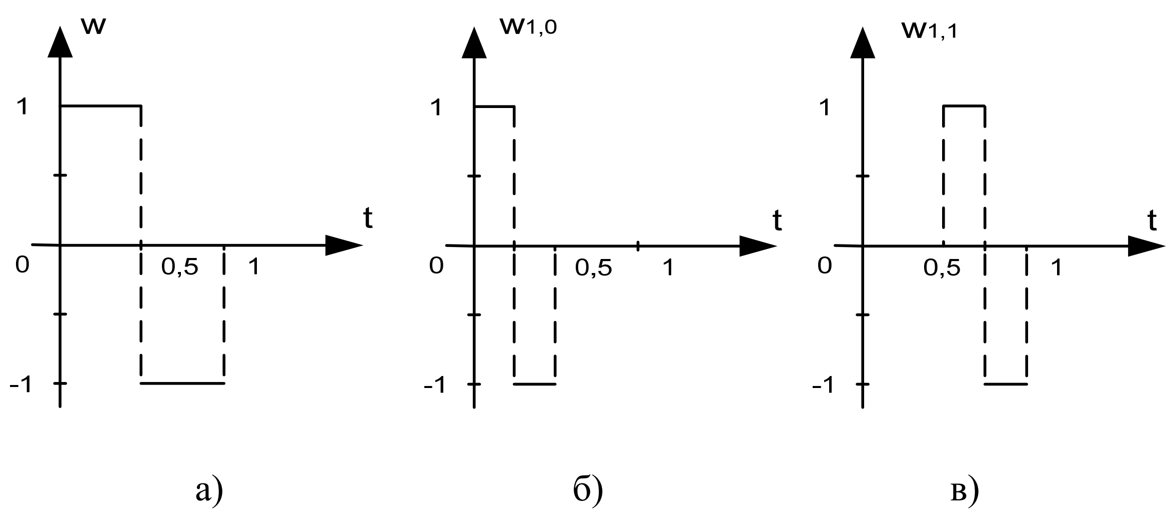


Рисунок 2.2 – Вейвлети Хаара

Функція $w(t)$ називається функцією Хаара (рисунок 2.2 а) одиничного масштабу і нульового зміщення (материнський вейвлет Хаара) – це функція, що дорівнює +1 на інтервалі [0; 1/2) і -1 на інтервалі [1/2; 1). А функції, що представлені на рисунках 2 б та 2 в можуть бути отримані перетвореннями (стисненням та розтягуванням відносно осі абсцис) і зсувом аргументу функції $w(t)$, що визначається формулою:

$$w(t)=\begin{cases}1\,,\ t\in[0,1/2),\\ -1,\ t\in[1/2,1),\\ 0,\ \ t\notin[0,1).\end{cases} \tag{2.10}$$

У додатку А знаходиться код програмної реалізації методу КС на базі методу Канні та реалізація критерію Прета для його дослідження.

Приклад тестового зображення представлено на рисунку 2.3. Дане зображення підготоване з врахуванням особливостей предметної області. Воно відтворює набір характерних ознак реальних об'єктів у областях чорного, білого та сірого кольору. Це зображення є репрезентативним, тобто за показниками продуктивності і точності його обробки можна судити про середню якість опрацювання реальних об'єктів.

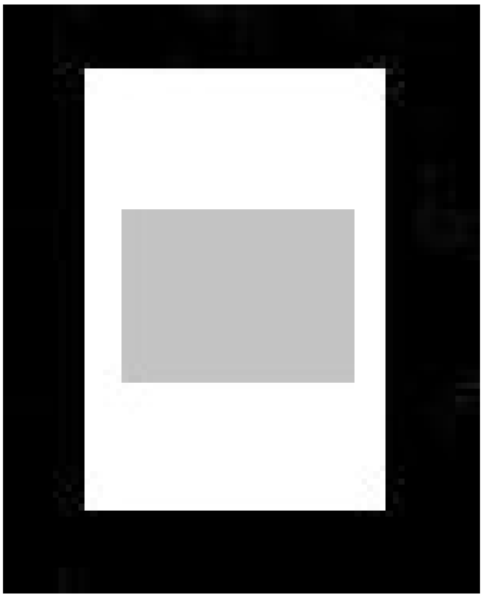


Рисунок 2.3 – Тестове зображення

На рисунку 2.4, для наочного прикладу, представлена стрічка зображення, графік інтенсивності її пікселів ($I_{61,j}$); та сама стрічка зображення із накладеним на неї вейвлетом Хаара, графік інтенсивності її пікселів ($WX_{61,j}$), а також графік, що утворюється після встановлення порогу $WX'_{61,j}$.

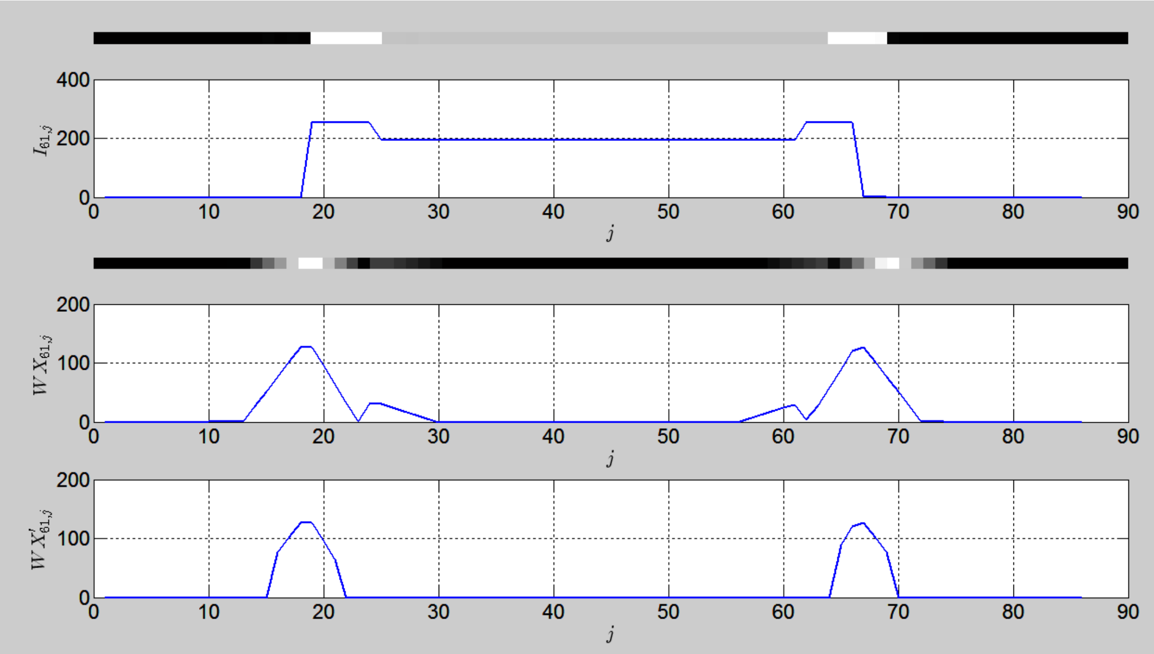


Рисунок 2.4 – Стрічка зображення та графіки інтенсивності

На рисунку 2.5 представлено зображення, що отримуємо після накладання вейвлетів на всі пікселі зображення загалом: а) – по рядках, в) – по стовпцях; та встановлення порогу (б, г).

Остаточним результатом на першому рівні ієрархії отримаємо зображення, яке представлено на рисунку 2.6 а), а на наступному рівні ієрархії – рисунку 2.6 б). Класичним методом Канні контури виділені на рисунку 2.6 в).

Запропонований метод досліджувався за критерієм Прета [26] на чорно-білому перепаді. Цей критерій враховує три види помилок: розриви контуру, локальні зміщення контуру та розмитість контуру. Обчислюється за формулою:

$$R = \frac{\sum_{i=1}^{I_A} \frac{1}{1+\alpha d_i^2}}{\max(I_i, I_A)}, \tag{2.11}$$

де I_i - число елементів у ідеальному контурі, I_A - число елементів у реальному контурі, d_i - величина зміщення i-того елемента виділеного контуру по нормалі до лінії ідеального контуру; α - масштабний множник, що забезпечує відношення між «штрафами» за зміщений і розмитий контури ($\alpha = 1/9$). Значення критерію нормалізоване так, що $R = 1$ буде тільки для точно виділеного контуру. У всіх інших випадках значення R буде меншим за 1.

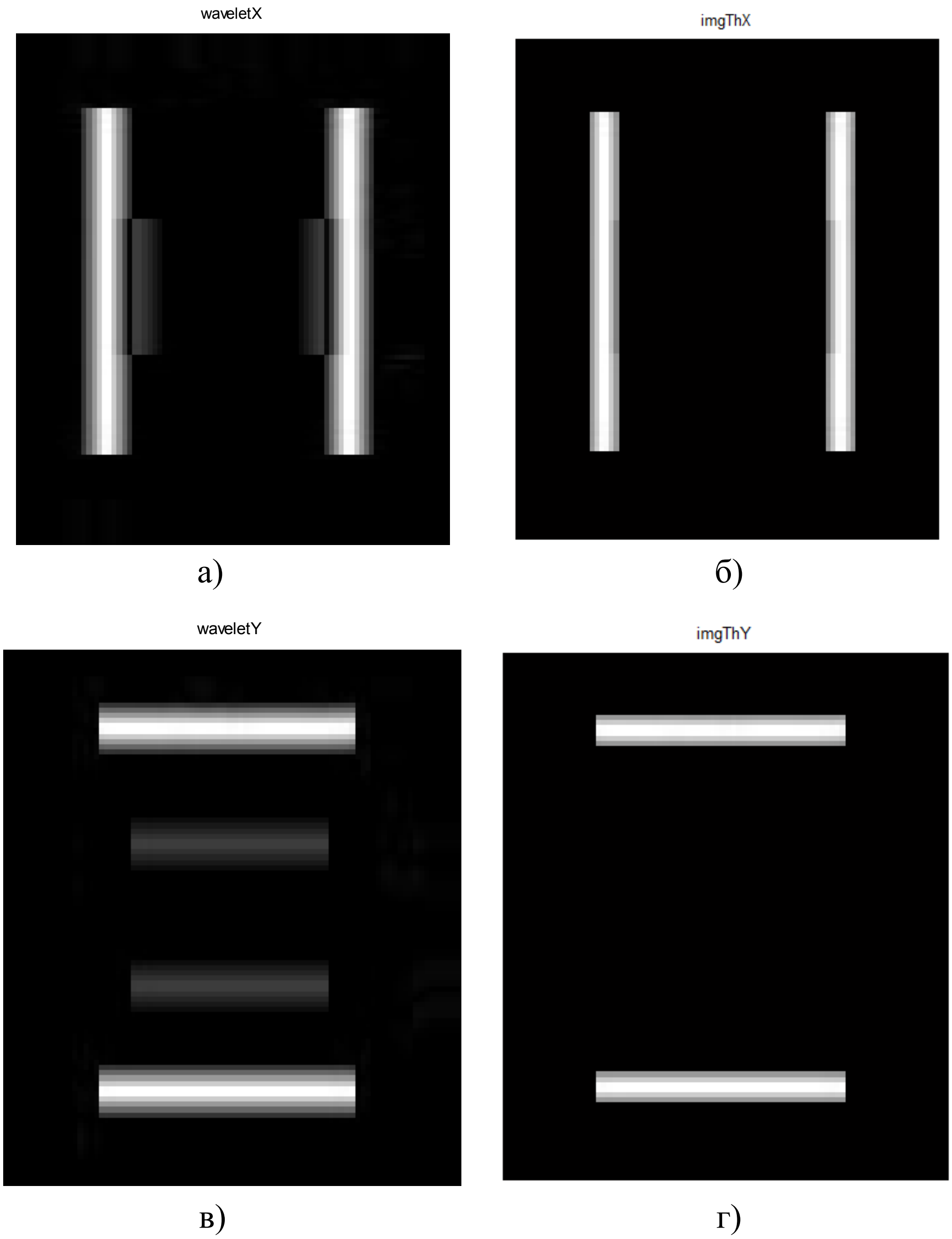
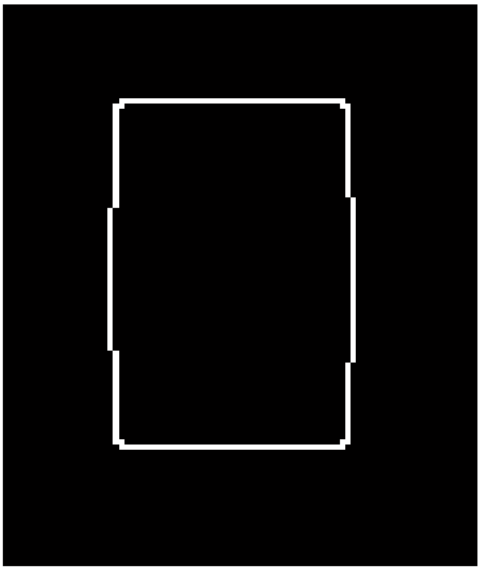
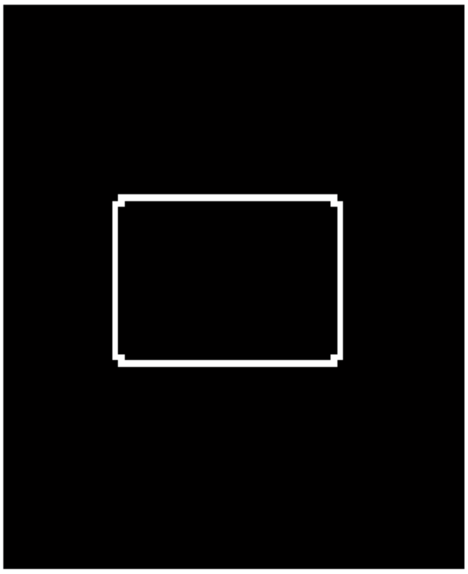


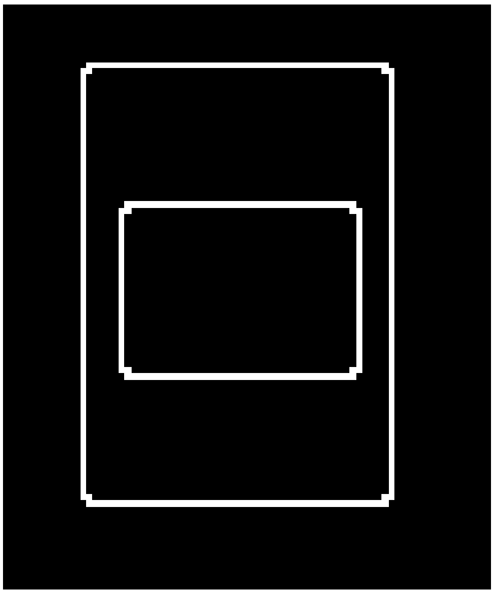
Рисунок 2.5 – Зображення, що отримуємо після накладання вейвлетів: а) – по рядках, в) – по стовпцях; та встановлення порогу (б, г)



а)



б)

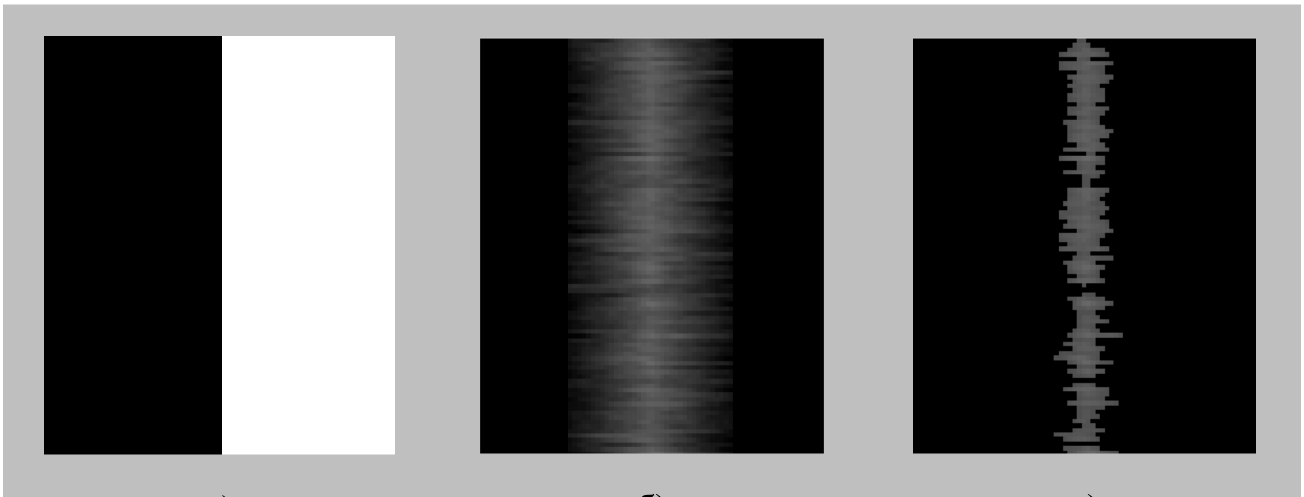


в)

Рисунок 2.6 – Результати виділення контурів запропонованим методом на першому рівні ієрархії (а) та на другому рівні ієрархії (б); методом Канні (в)

При дослідженні описаного методу використовувалось тестове зображення, що складалось із 92x76 пікселів (рисунок 2.7 а). В центрі цього зображення знаходиться вертикально орієнтований чорно-білий перепад яскравості. Для наочності зображення представлене на сірому фоні. На зображення був накладений Гаусівський шум із різними відношеннями сигнал/шум.

На рисунку 2.7 б наведено зображення, яке отримуємо після накладання вейвлету (застосування формули 2.2) на зображення із відношенням сигнал/шум 5 довжиною вейвлета $n = 15$ по осі *OX*. А на рисунку 2.7 в – зображення, яке отримуємо після порогової фільтрації (застосування формули 2.5, величина порогу $P = 0{,}3$).



а) б) в)

Рисунок 2.7 – Тестове зображення (а) та проміжні зображення (б-в)

На рисунку 2.8 представлені зашумлені зображення та результати обробки даних зображень запропонованим методом. Для всіх випадків довжина вейвлету $n = 15$, поріг фільтрації $P = 0{,}3$.

А на рисунку 2.9 наведені графіки залежності критерію Прета від відношення сигнал/шум, які дозволяють провести порівняльну оцінку запропонованого методу із відомими, зокрема проведено порівняння з методом Канні, на основі якого він побудований. Як видно із наведених графіків, запропонований метод дає кращі результати за метод Канні для всіх відношень сигнал/шум при довжині вейвлета більшій за 5. Найбільший виграш отримано при низьких відношеннях сигнал/шум. А при малих значеннях довжини вейвлету результати подібні.

Метод був апробований на відеоінформації, отриманій з комп’ютерних систем відеонагляду. У відповідність обличчю людини ставилась послідовність контурів (рисунок 2.10).

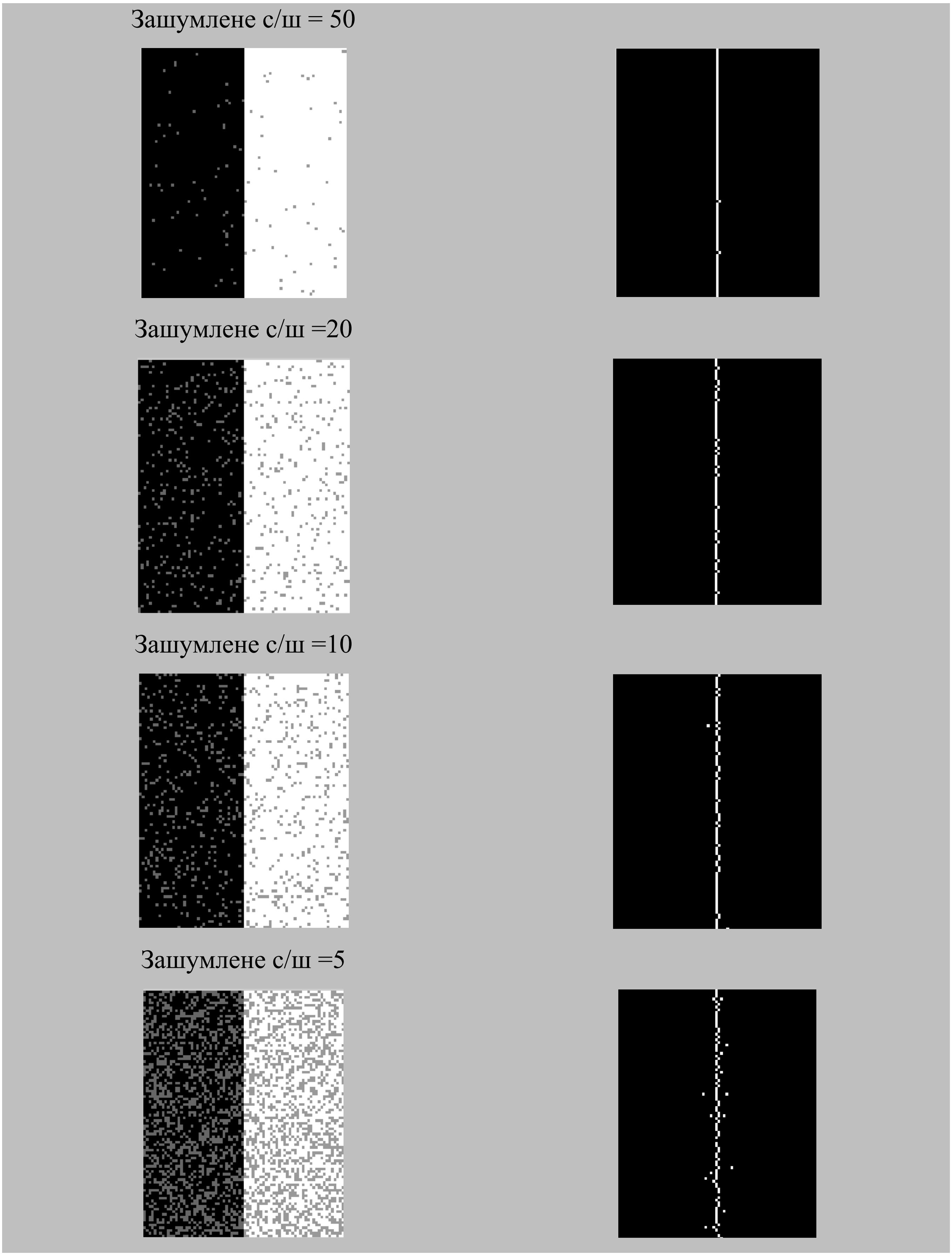


Рисунок 2.8 – Результати обробки чорно-білого перепаду для різних відношень сигнал/шум

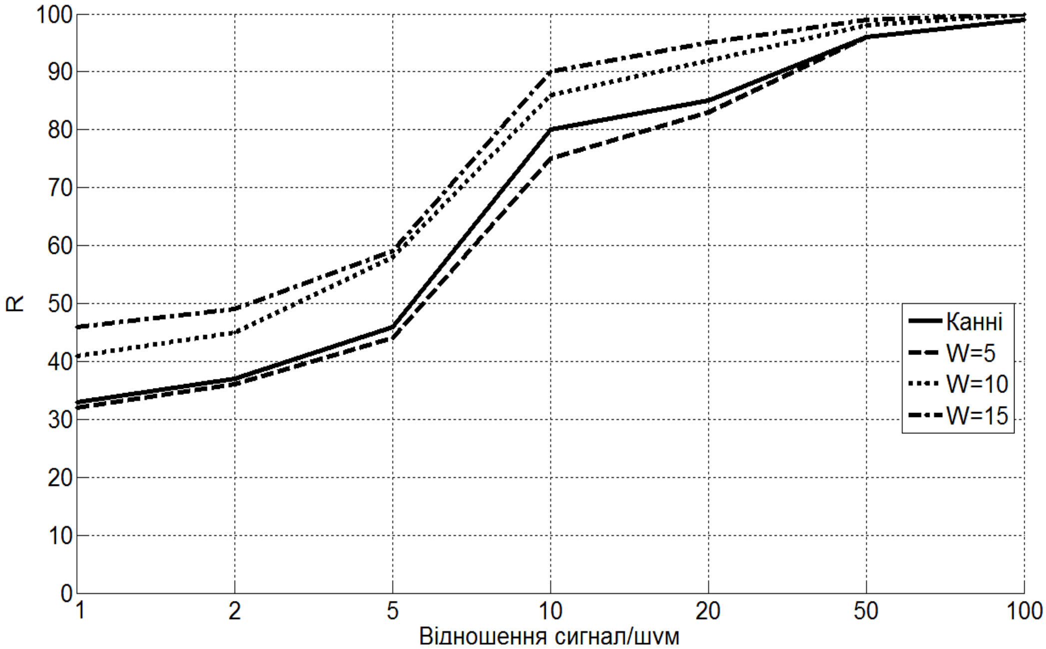


Рисунок 2.9 – Графік залежності величини критерія Прета від відношення сигнал/шум чорно-білого перепаду операторами виділення контурів

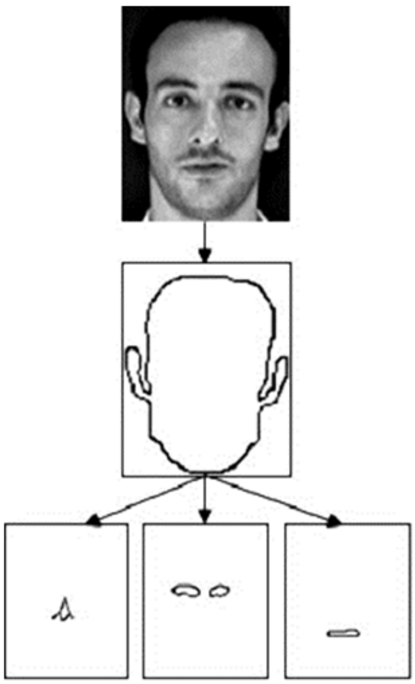


Рисунок 2.10 – Ієрархічне представлення контурів обличчя

Отже, у даному підрозділі розроблено та експериментально досліджено метод КС на базі методу Канні з використанням вейвлет-функцій для підкреслення перепадів інтенсивності зображення. Експериментальні дослідження показали, що розроблений метод дає кращі результати за критерієм Прета у 1,2 – 1,5 разів у порівнянні з методом Канні при співвідношенні сигнал/шум меншому за 10, а довжині вейвлета більшій за 5.

2.2. Аналіз функції кривизни

Основним алгоритмічним апаратом, що використовується для виділення характерних точок контуру вважається аналіз функції кривизни [24, 28].

Розглянемо безперервну замкнуту криву, побудовану в комплексній площині (рисунок 2.11) для котрої положення точки на периметрі задається її координатою $z(s)$, залежною від довжини s [26, 36].

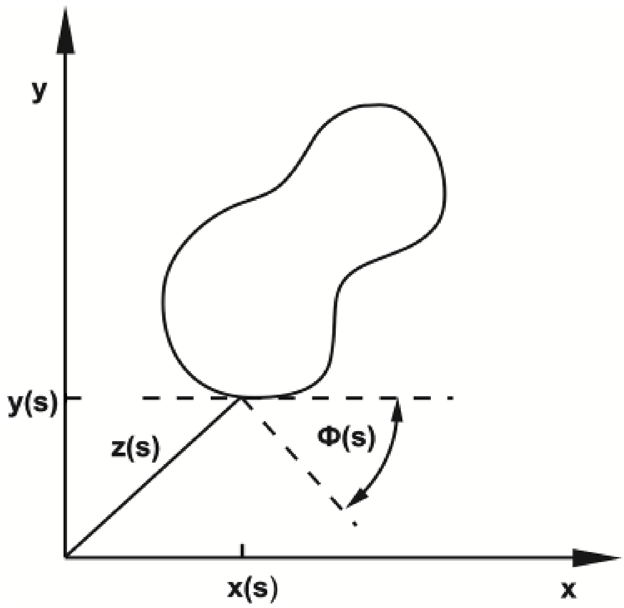


Рисунок 2.11 – Визначення кривизни

Комплексну функцію $z(s)$ можна представити у вигляді її дійсної $x(s)$ та уявної $y(s)$ складових:

$$z(s) = x(s) + iy(s) \tag{2.12}$$.

Кут нахилу дотичної, показаний на рисунку 2.14, визначається виразом:

$$\Phi(s) = arctg\left[\frac{dy(s)/ds}{dx(s)/ds}\right], \tag{2.13}$$

а кривизна являє собою дійсну функцію

$$k(s) = d\ \Phi\ (s)/ds\,. \tag{2.14}$$

Знаючи функцію кривизни, координати точок $x(s)$, $y(s)$ можна отримати за формулами:

$$z(s) = x(0) + \int_0^s k(\alpha)\cos[\Phi(\alpha)]d\alpha\,, \tag{2.15}$$

$$y(s) = y(0) + \int_0^s k(\alpha)\sin[\Phi(\alpha)]d\alpha\,, \tag{2.16}$$

де $x(0)$ і $y(0)$ – координати початкової точки.

Якщо форма має різкі злами, наприклад, прямокутні, функція кривизни в точках розриву виявляється невизначеною. Ці аналітичні труднощі можна обійти, використовуючи функцію форми запропоновану в [26]:

$$\theta(s) = \int_0^s k(\alpha)d\alpha - \frac{2\pi s}{p}\,, \tag{2.17}$$

Для дискретного випадку кривизна визначається формулами [36]:

$$z(s_i) = x(s_i) + iy(s_i)\,, \tag{2.18}$$

$$\Phi(s_i) = arctg\left[\frac{y(s_i) - y(s_{i-1})}{x(s_i) - x(s_{i-1})}\right], \tag{2.19}$$

$$k(s_i) = \Phi(s_i) + \Phi(s_{i-1}). \tag{2.20}$$

де s_i і-тий елемент дуги.

Для полярної системи координат кривизна визначається наступним чином: нехай $(x_i, y_i) \in X$ – впорядкована множина точок контуру зображення в декартовій системі координат, тоді:

$$\overline{x} = \frac{1}{|X|}\sum_{i=1}^{|X|} x_i, \qquad \overline{y} = \frac{1}{|X|}\sum_{i=1}^{|X|} y_i, \tag{2.21}$$

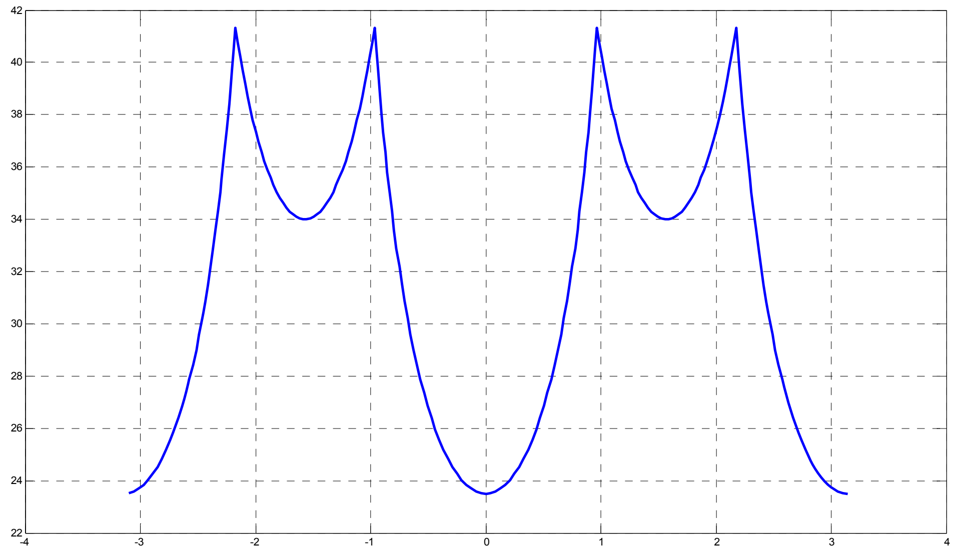
де $\overline{x}$, $\overline{y}$ – середня точка зображення контуру, а $|X|$ – кількість точок у множині X.

Функцію кривизни можна побудувати за формулою:

$$\rho_i = \sqrt{\overline{x_i}^2 + \overline{y_i}^2}, \tag{2.22}$$

де $\overline{x_i} = x_i - \overline{x}$, $\overline{y_i} = y_i - \overline{y}$, при чому, величина кривизни ρ_i відповідає куту $\theta_i = arctg\left(\frac{\overline{y_i}}{\overline{x_i}}\right)$.

Приклади функції кривизни для контурів зображень представлені на рисунку 2.12. На графіку по осі абсцис встановлено значення кута, який знаходиться в діапазоні $(-\pi;\pi)$, а по осі ординат – значення функції кривизни ρ_i.



а)

б)

Рисунок 2.12 – Приклад функції кривизни:

а) для прямокутника; б) для контуру голови

В міру диференціальної природи, функція кривизни має низьку завадостійкість: будь-які максимуми і мінімуми відзначаються як ХТ, за рахунок чого виділяється дуже велика кількість таких точок. Тому, для виходу з даної ситуації, запропоновано використати вейвлети Хаара, як найпростіші вейвлети, які добре зарекомендували себе в практичних завданнях обробки дискретних сигналів.

2.3. Метод вейвлет-перетворення функції кривизни для виділення характерних точок контуру

Для роботи методу вейвлет-перетворення функції кривизни для виділення ХТ контуру потрібно виконати прослідковування контуру та записати у вигляді масиву координат точок контуру. Піксель з якого починається обхід контуром об’єкта вибирається довільно, наприклад крайній лівий верхній піксел, що належить об’єкту. Сусідній контурний піксель, що належить контуру об’єкта отримуємо за допомогою алгоритму „жука” [29]. Прослідковування контуру здійснюється на основі таких правил: якщо значення поточного пікселя дорівнює одиниці, то поточний піксель належить контуру і проводиться поворот ліворуч; якщо значення поточного пікселя дорівнює нулю, поточний піксель не належить контуру об’єкта і проводиться поворот праворуч. Алгоритм зупиняє свою роботу, якщо повернувся в стартову точку.

Після того як контур прослід кований, виконуємо наступні кроки [46]:

1) Формується функція кривизни для прослідкованого контуру, як диференціальна функція координат контуру. Для декартової системи координат за формулами (2.18) – (2.20), а для полярної системи координат за формулами (2.21) – (2.22).

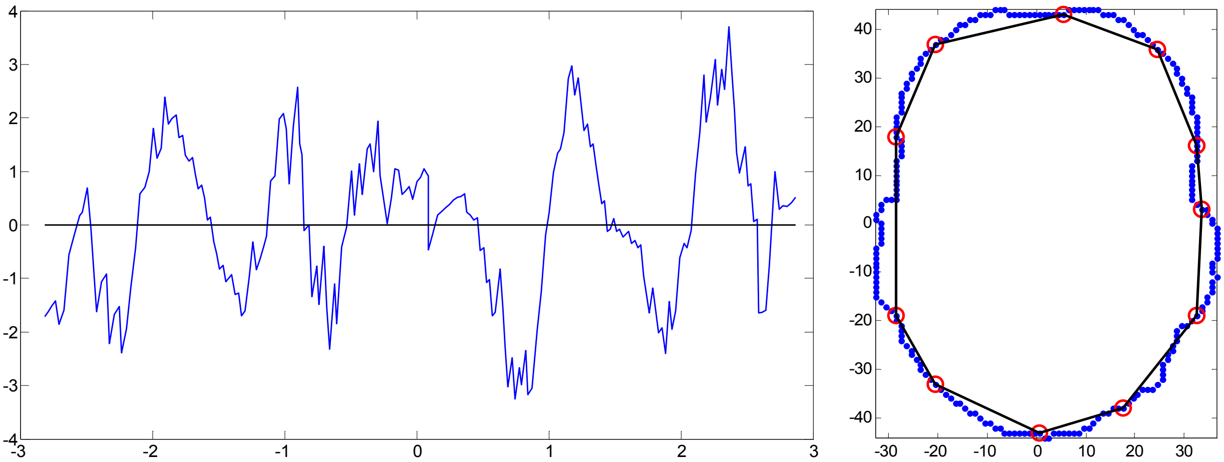
2) Здійснюється вейвлет-перетворення функції кривизни в базисі Хаара, яке полягає в лінійному перетворенні функції ρ парної розмірності 2π в іншу H за наступними співвідношеннями:

$$H_j = \sum_{i=j-a}^{j-1} \rho_i \cdot 1 + \sum_{i=j+1}^{j+a} \rho_i \cdot (-1), \qquad (2.23)$$

де $j \in [-\pi + a; \pi - a)$, $2a$ – довжина вейвлету Хаара.

До ХТ належать ті точки, які знаходяться на перетині графіка функції $H(\rho)$ з віссю ox.

На рисунку 2.13 а) зображена функція, яка утворилась внаслідок накладання вейвлета Хаара при $a = 10$ на функцію кривизни з рисунку 2.12 б згідно формули 2.23. Характерні точки на представленому графіку знаходяться на перетині графіка з віссю абсцис і на рисунку 2.13 б) перенесені на контур голови та послідовно сполучені.



а) б)

Рисунок 2.13 – Функція, яка утворилась внаслідок накладання вейвлету Хаара (а) та ХТ перенесені на контур голови (б)

3) Якщо рівень деталізації не задовольняє, то зменшуємо довжину вейвлета та повторюємо описану вище процедуру.

Узагальнена схема роботи методу представлена на рисунку 2.14.

В залежності від значення параметра a буде виділятись різна кількість ХТ: чим менше значення a – тим більше буде ХТ. Приклади наведені у таблиці 2.1. А у таблиці 2.2 представлені наочні результати роботи різних методів виділення ХТ.

Для порівняння результатів опрацювання зображень методами виділення ХТ контуру запропоновано використати три критерії:

1) ймовірність правильного виявлення;

2) ймовірність хибного виявлення;

3) похибка в оцінці координат, що базуються на результатах модельної задачі.

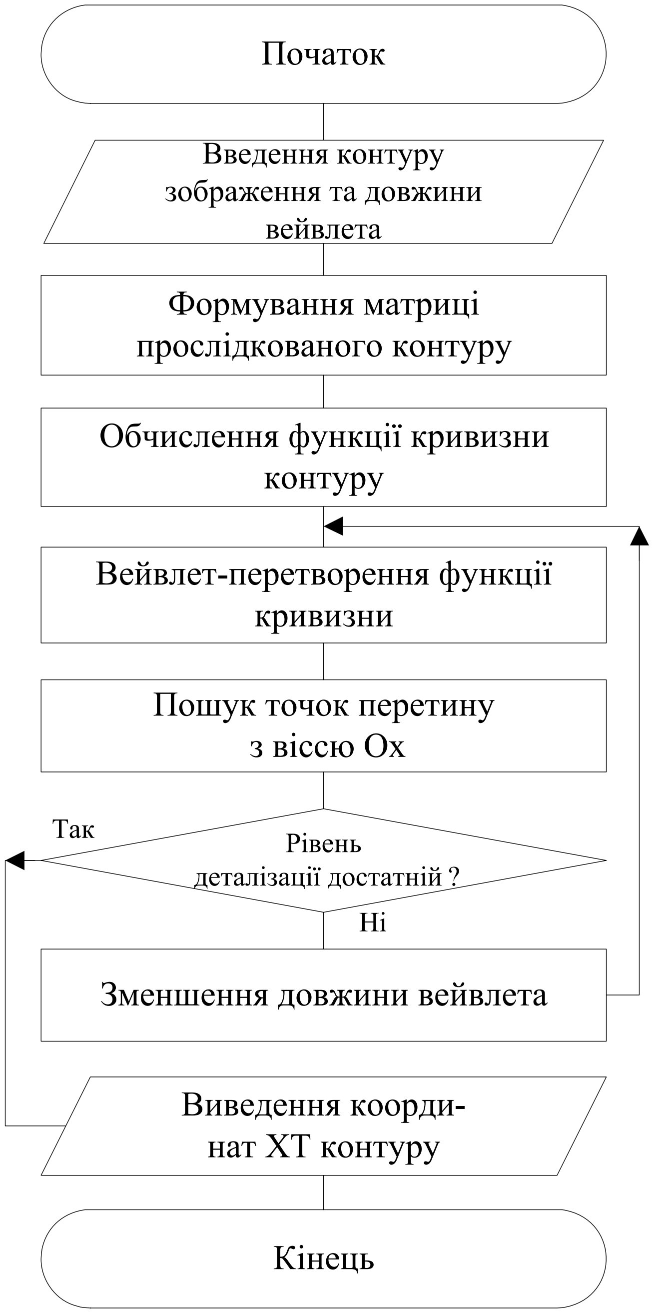
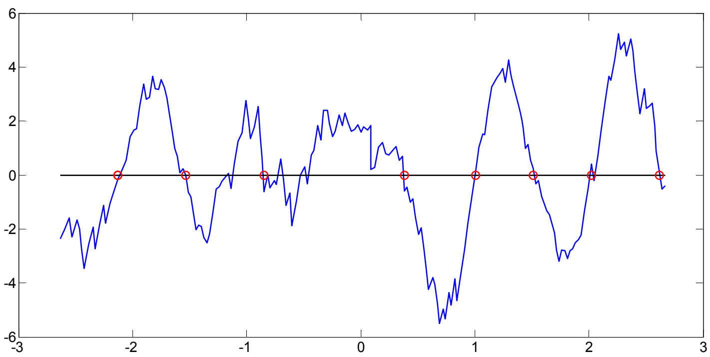
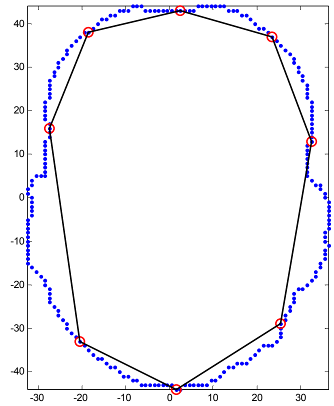
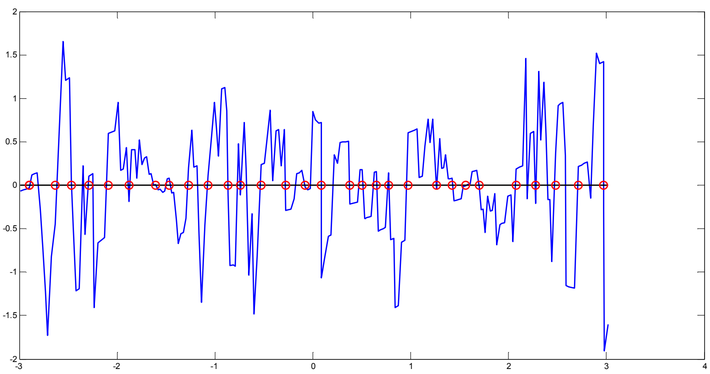
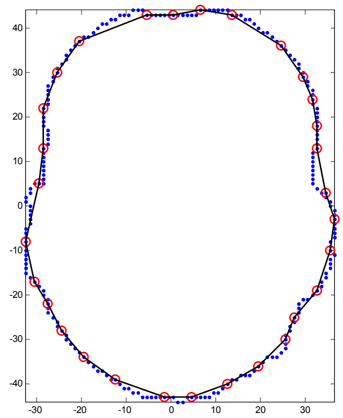
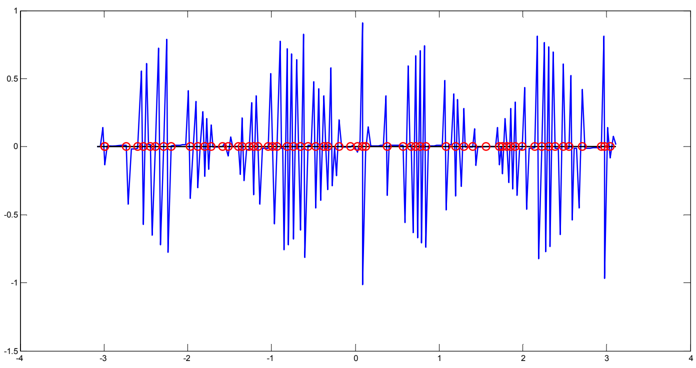
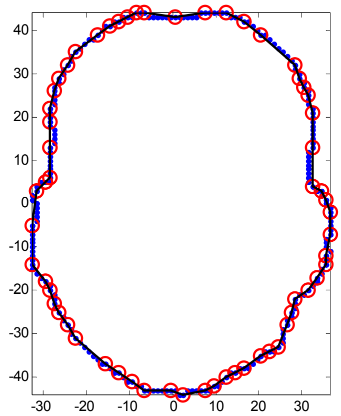


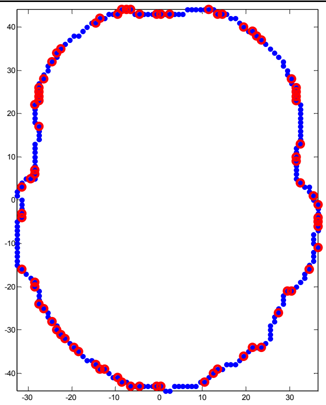
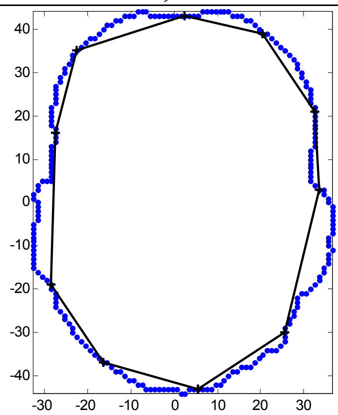
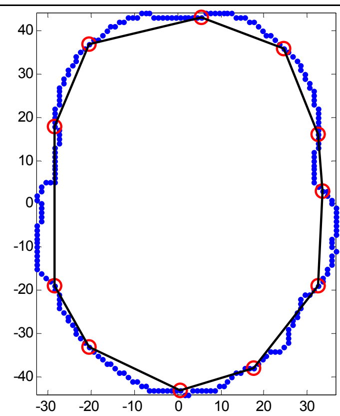
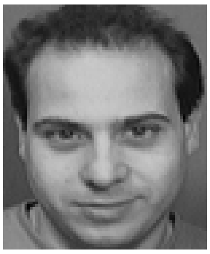
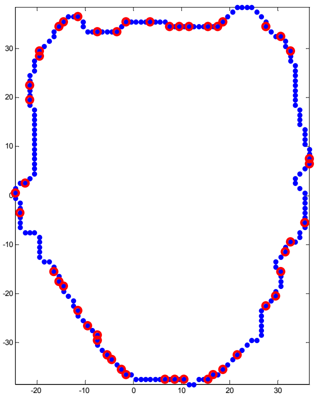
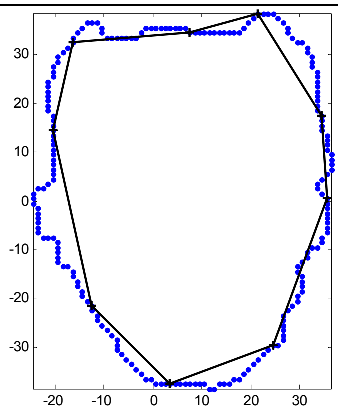
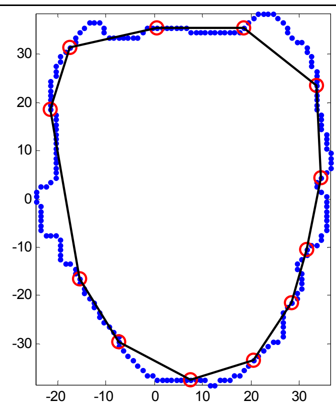
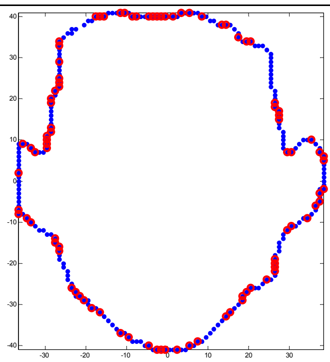
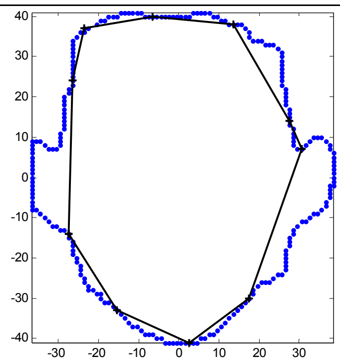
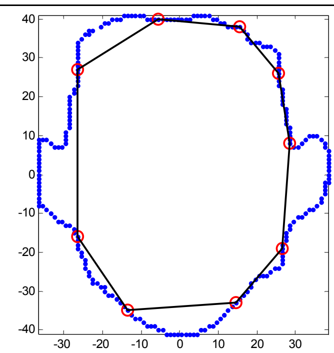
Рисунок 2.14 – Схема методу виділення ХТ

Порівняння методів базується на результатах модельної задачі та полягає у виявленні ХТ на контурах зображень простих геометричних фігур, оскільки для таких фігур наперед відоме розміщення ХТ – це будуть вершини многокутників. Тому, кількість ХТ в розв’язку модельної задачі буде дорівнювати кількості вершин многокутника [33].

Таблиця 2.1 – Залежність кількості виділених ХТ від довжини вейвлету Хаара

Значення параметра *a* та кількість отриманих ХТ	Приклади отриманих функцій після вейвлет-перетворення функції кривизни в базисі Хаара	Контур зображення із нанесеними та послідовно сполученими ХТ
a = 16, ХТ = 8		
a = 4, ХТ = 30		
a = 1, ХТ = 61		

Таблиця 2.2 – Порівняння роботи методів виявлення характерних точок

Вхідне зображення	Виділення характерних точок контуру		
	Диференціальний метод	Інтерполяційним метод n=4, Δ=7	Запропонованим метод a= 10
 К-сть точок контуру - 233	 К-сть ХТ – 78	 к-сть ХТ – 10	 к-сть точок – 11
 К-сть точок контуру - 213	 К-сть ХТ – 49	 к-сть ХТ – 9	 К-сть точок – 12
 К-сть точок контуру - 243	 К-сть ХТ – 92	 к-сть точок – 10	 К-сть точок – 9

Ймовірність правильного виявлення обчислюється як відношення кількості значень, що відповідають заданій умові до кількості значень які б мали відповідати заданій умові. В нашому випадку це відношення кількості ХТ, що співпадають з модельними ХТ до кількості модельних ХТ:

$$P = \frac{C_{quality}}{K} \tag{2.24}$$

де P – значення ймовірності правильного виявлення, $C_{quality}$ – кількість ХТ, виявлених алгоритмом, що співпадають з модельними ХТ, K – кількість ХТ, розв'язок модельної задачі.

Значення P знаходиться в межах $P \in [0..1]$, та чим більшим є значення, тим краще виділені ХТ. Значення 1 отримаємо в тому випадку, коли виділені алгоритмом ХТ співпадуть із всіма модельними ХТ. А значення 0 буде в тому випадку, коли жодна ХТ виділена алгоритмом не співпаде з модельною ХТ.

Проте, даний критерій не враховує того факту, що за допомогою методу можна виділити набагато більше ХТ, ніж є модельних ХТ та вони не будуть співпадати з модельними ХТ. Тому запропоновано використати ще ймовірність хибного виявлення та похибку в оцінці координат.

Ймовірність хибного виявлення ХТ обчислюється за формулою:

$$Q = \frac{C_{total} - C_{quality}}{C_{total}} \tag{2.25}$$

де Q – ймовірність хибного виявлення, C_{total} – загальна кількість виявлених алгоритмом ХТ.

Значення Q буде знаходитись в межах $Q \in [0..1]$, та чим більшим буде значення, тим гірші результати роботи методу. Значення 1 буде в тому випадку, коли жодна з виділених алгоритмом ХТ не буде співпадати з модельними ХТ. А значення 0 буде в тому випадку, коли кількість та всі виділені алгоритмом ХТ будуть співпадати з модельними ХТ.

Похибка в оцінці координат полягає в тому, щоб обчислити середнє арифметичне зміщення між модельними ХТ контуру та найближчими до них ХТ виділеними алгоритмом. Обчислюється за формулою:

$$E = \frac{\sum_{i=1}^{K} \left\| (x_{model,i} ; y_{model,i}) - (\hat{x}_i ; \hat{y}_i) \right\|}{K}, \tag{2.26}$$

де $(\hat{x}_i , \hat{y}_i) = \underset{(x,y)\in N}{\arg\min} \left\| (x, y) - (x_{model,i} , y_{model,i}) \right\|$, N – множина всіх характерних точок, виявлених алгоритмом; $(x_{model,i} , y_{model,i})$ – i-та характерна точка, розв’язок модельної задачі. Чим менше значення E тим точніше виявлені ХТ.

Порівняння методів виділення ХТ здійснювалось на оригінальному зображенні та в умовах впливу рівномірно розподіленого адитивного гаусівського шуму, що характеризується рівномірною спектральною щільністю, нормально розподіленим значенням амплітуди і адитивним способом впливу на сигнал (шуму підсумовується з корисним сигналом), це найпоширеніший вид шуму, який використовується для розрахунку і моделювання.

У додатку Б знаходиться код реалізованих в пакеті прикладних програм Matlab у вигляді виконуваних m-скриптів диференціальний метод, інтерполяційний метод, запропонований метод на основі вейвлет-аналізу функції кривизни та реалізація критеріїв порівняння даних методів.

На рисунку 2.15 графічно представлено результати роботи методів реалізованих в пакеті прикладних програм Matlab у вигляді виконуваних m-скриптів: (а) і (г) інтерполяційного з параметрами довжини інтерполяційного відрізка (n=2) та відстані від проміжної точки контуру до інтерполяційного відрізка (DELTA=2), (б) і (д) диференціального та (в) і (е) запропонованого на основі вейвлет-аналізу функції кривизни (з довжиною вейвлета a=10). На рисунках кружечками позначено точки контуру, які виділено як характерні відповідним методом. Такі точки послідовно сполучені для наочної оцінки контуру фігури, що утворюється.

На рисунку 2.16 наведено результати порівняння методів виділення ХТ з оригінальної відеоінформації та в умовах впливу рівномірно розподіленого адитивного гаусівського шуму за критеріями: ймовірності правильного виявлення (P), ймовірності хибного виявлення (Q) та похибки в оцінці координат (E).

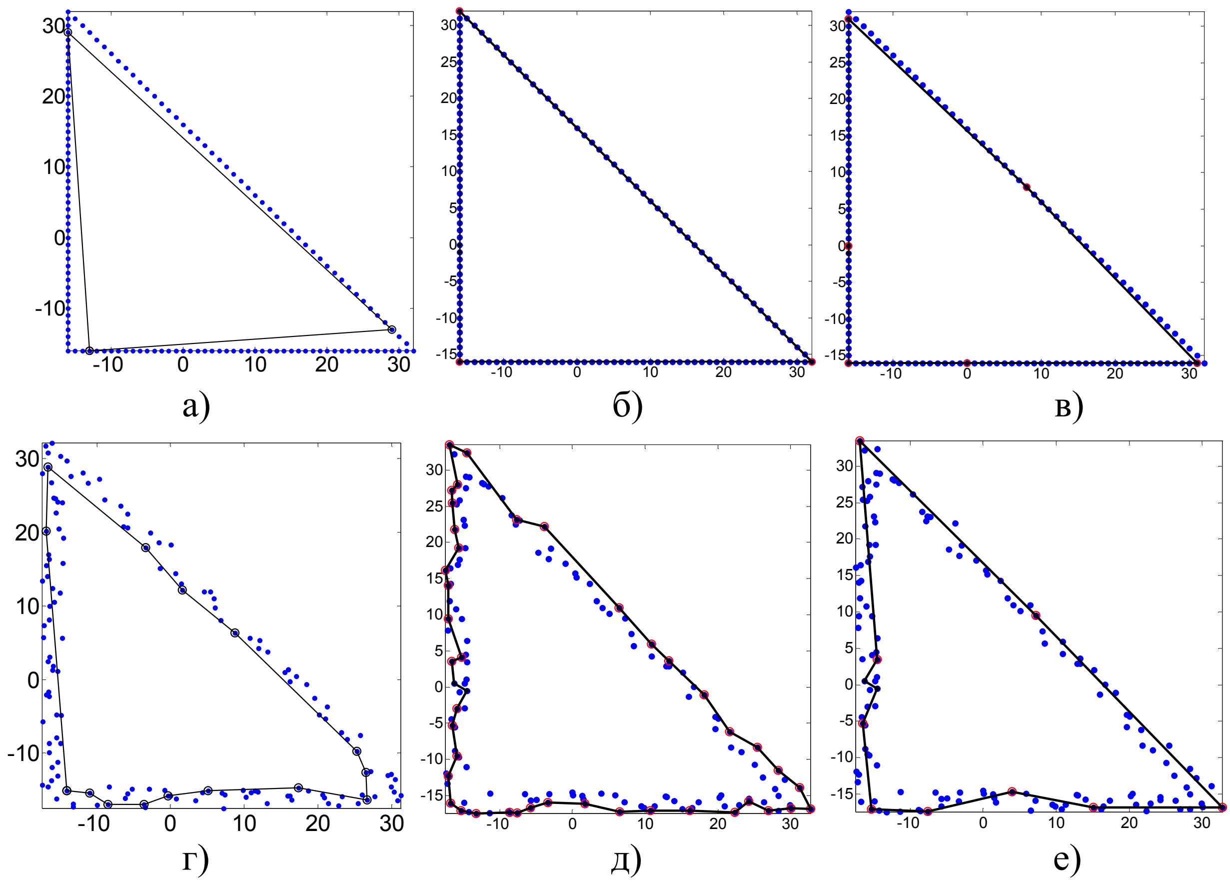


Рисунок 2.15 – Графічне представлення результатів роботи методів виділення ХТ контуру на незашумлених зображеннях (а, б, в) та в умовах впливу рівномірно розподіленого адитивного гаусівського шуму (г, д, е) інтерполяційним, диференціальним та методом на основі вейвлет-аналізу функції кривизни відповідно

Як видно із представлених графіків, диференціальний метод на зображенні без шуму показує найкращі результати за всіма критеріями, а інтерполяційний – найгірші, оскільки відбувається незначний зсув виявлених алгоритмом ХТ від вершин багатокутника (модельних ХТ).

На тому ж зображенні, що було додатково спотворено шумами, диференціальний метод та метод на основі вейвлет-аналізу функції кривизни показують однакові результати за критеріями ймовірності правильного виявлення та похибки в оцінці координат. За критерієм ймовірності хибного виявлення кращий результат показує метод на основі вейвлет-аналізу функції кривизни.

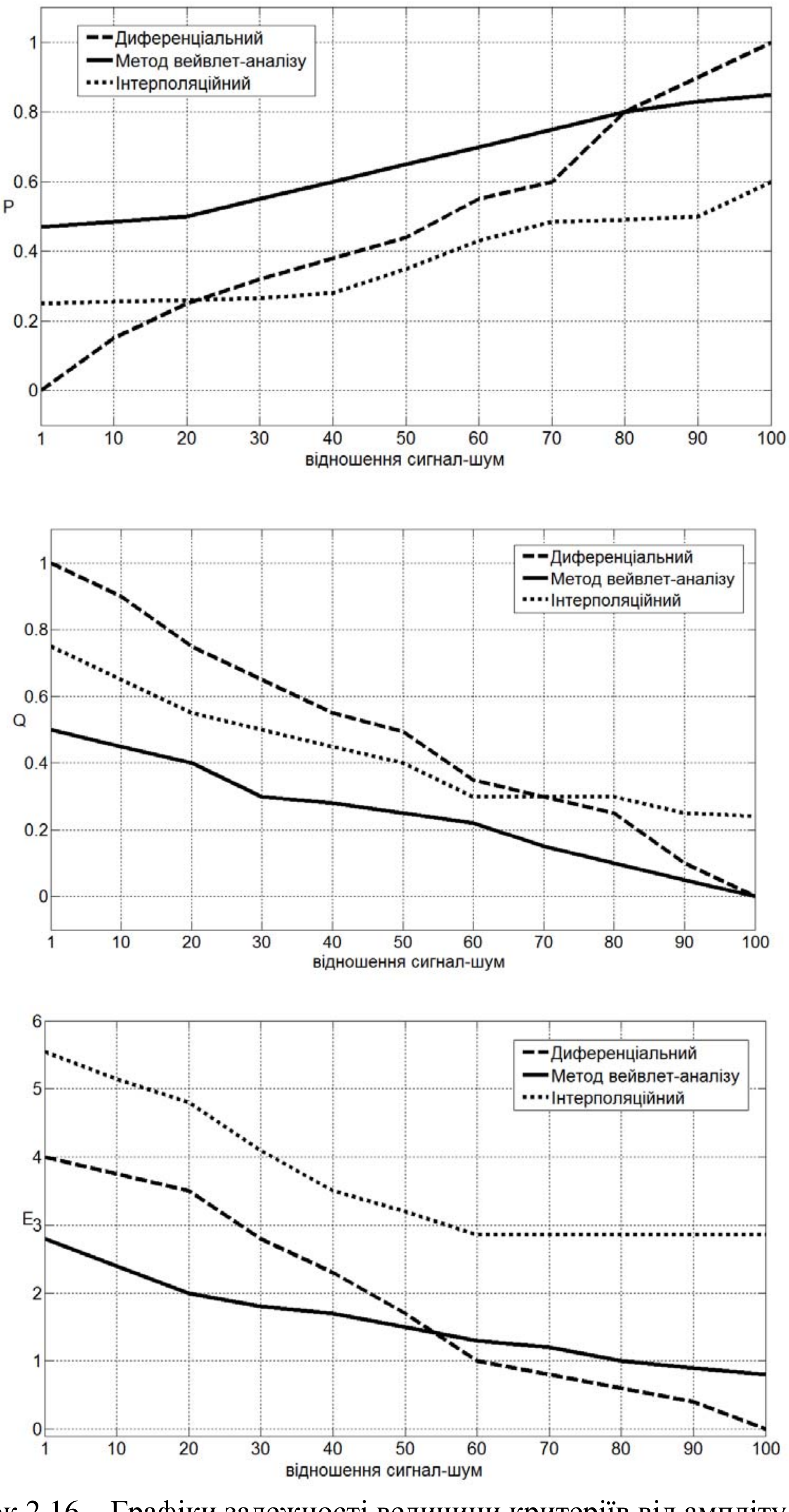


Рисунок 2.16 – Графіки залежності величини критеріїв від амплітуди шуму

Дослідження показали, що інтерполяційний метод має низьку точність (при будь-якому відношенні сигнал/шум), бо характерні точки, виділені ним, зсунуто відносно модельних характерних точок приблизно на довжину інтерполяційного відрізка. Диференціальний метод має низьку завадостійкість (у 1,2-2 рази гіршу, ніж розроблений метод при відношенні сигнал/шум меншому за 70), оскільки після зашумлення зображення трактує надто велику кількість точок характерними.

Метод на основі вейвлет-аналізу функції кривизни показує найкращі результати:

- за критерієм ймовірності правильного виявлення (P) при відношенні сигнал/шум меншому за 80;

- ймовірності хибного виявлення (Q) для будь-якого відношенні сигнал/шум;

- похибки в оцінці координат (E) при відношенні сигнал/шум меншому за 50.

Висновки до розділу 2

У даному розділі розроблено та досліджено метод та базову ІТ контурної сегментації на базі методу Канні; метод та базову ІТ виділення ХТ контуру з регульованою деталізацією.

Вдосконалено метод КС Канні за рахунок доозначення оригінального зображення, застосуванням вейвлету Хаара для згортки рядка зображення, пороговому фільтруванні та аналізі на екстремуми функції інтенсивності. Цей метод дає змогу успішно виділяти контури зашумлених зображень та представляти, за необхідності, контурний препарат у вигляді ієрархічної структури (об’єкт-підоб’єкт).

Метод КС на базі методу Канні досліджувався для вейвлетів Хаара різної довжини та порівнювався із методом Канні за критерієм Прета на чорно-білому перепаді при різних співвідношеннях сигнал/шум. Експериментальні дослідження показали, що завадостійкість розробленого методу КС за критерієм Прета у 1,2 – 1,5 разів вища за метод Канні.

У результаті аналізу методів виділення ХТ контуру розроблено метод виділення ХТ з регульованою деталізацією в основі якого лежить вейвлет-аналіз функції кривизни, зокрема з допомогою вейвлету Хаара, оскільки в нього низька обчислювальна складність, що значно спрощує його апаратну реалізацію та підвищує оперативність. Експериментально досліджено цей метод за критеріями: ймовірність правильного виявлення (P), ймовірність хибного виявлення (Q) та похибка в оцінці. Координат (E), та виконано порівняння із відомими методами такими як диференціальний та інтерполяційний. Для порівняння використано прості геометричні фігури та модельну задачу, як еталон правильного розміщення ХТ (в простих геометричних фігур це будуть вершини многокутників), та проведено аналіз роботи цих методів на звичайних та зашумлених зображеннях.

Дослідження показали, що в інтерполяційному методі точність нижча у 1,5 – 3 рази за розроблений метод (бо ХТ виділені інтерполяційним методом зсунуті відносно модельних ХТ приблизно на довжину інтерполяційного відрізка), диференціальний метод має нижчу завадостійкість у 1,2 – 2 рази (оскільки після зашумлення зображення виділяє занадто велику кількість точок як характерних), а метод на основі вейвлет-аналізу функції кривизни показує найкращі результати за всіма критеріями при відношенні сигнал/шум меншому за 50.

Основні результати розділу опубліковані в працях [33, 34, 36, 37, 38, 42, 44, 45, 46].

РОЗДІЛ 3
МЕТОД ІЄРАРХІЧНОЇ СТРУКТУРНО-СТАТИСТИЧНОЇ ІДЕНТИФІКАЦІЇ ОБ'ЄКТІВ ЗА ХАРАКТЕРНИМИ ТОЧКАМИ КОНТУРУ З ДЕТАЛІЗАЦІЄЮ ЯКА РЕГУЛЮЄТЬСЯ

3.1. Метод ідентифікації зображень за характерними точками контуру

Завдання автоматизованого або автоматичного розпізнавання просторових об'єктів відноситься до складних завдань комплексного типу [102]. Виділення ознак передбачає отримання деякого вектора ознак, що дозволяє стійко класифікувати об'єкт при заданих умовах. У більшості випадків до вектору ознак ставляться такі вимоги [7, 20]:

- завадостійкість;
- інваріантність до масштабу;
- інваріантність до руху в полі зору;
- інваріантність до повороту;
- висока обчислювальна ефективність в умовах обмежених обчислювальних ресурсів.

У задачах класифікації об'єкт, описаний вектором ознак, відноситься до певного класу.

Ідентифікаційні вектори можна будувати на основі:

- Статистичних ознак (коефіцієнт кореляції, моменти різних порядків, розподіл законів, матриці суміжності). Недоліками цих ознак є: значні обчислювальні витрати; невідомість закону розподілу; неоднозначність опису області зображення.
- Геометричних ознак. Вони поділяються на прості та похідні. До простих ознак відносять: периметр та площу фігури. До похідних ознак відносяться: середню довжину хорд, ексцентриситет та ін.

– Структурних ознак. Вони полягають в представленні як всього зображення, так і зображення окремого суб'єкта у вигляді сукупності деяких примірних геометричних елементів та їх відносин між собою.

– Нейросистемних ознак, в основі яких покладений принцип дії нейронних клітин головного мозку людини. Нейрон має кілька входів, кожен з яких має свою вагову функцію, а також вихід та блоку сумування.

Для ідентифікації запропоновано використати метод геометричних моментів-ознак (ГМО) з використанням інформації виключно про характерні точки контуру [7].

ГМО в евклідовій системі координат мають вигляд:

$$m_{\alpha\beta} = \int_0^{Lx}\int_0^{Ly} I(x,y)x^{\alpha}y^{\beta}dxdy \qquad (3.1)$$

де I - зображення, що обробляється, L_x, L_y - геометричні розміри зображення, α, β - порядок момента.

Використання моментів в якості ознак базується на наступній теоремі [83]: нескінченна послідовність моментів зображення $\{m_{\alpha\beta}\}(\alpha,\beta = 0,1,2,3,...)$ однозначно визначається функцією яскравості, і, навпаки, функція яскравості однозначно визначається послідовністю моментів $\{m_{\alpha\beta}\}(\alpha,\beta = 0,1,2,3,...)$.

Отже, обчисливши деякі моменти зображення, можна його ідентифікувати.

До переваг методу ГМО належать: згладжувальні якості моментів як інтегральних показників визначають високу стійкість методу; ознаки легко нарощуються до отримання стійкого розпізнавання.

При розпізнаванні методом геометричних моментів-ознак зазвичай використовують не самі моменти-ознаки, а розраховані на їх базі характеристики, що володіють властивістю інваріантності до масштабу і повороту. Інваріантність до повороту легше реалізується в полярній системі координат. Загальний вираз для обчислення ГМП в полярній системі координат має вигляд:

$$\mu_p = \int_0^{R_{\max}} \int_0^{2\pi} \rho^p I(\rho,\varphi) d\rho d\varphi \tag{3.2}$$

де $I(\rho,\varphi)$ - функція яскравості зображення в полярній системі координат; ρ,φ - полярні координати пікселів відносно центра ваги об'єкта; p - порядок моменту; $R_{\max}$ - відстань від центру ваги фігури до максимально віддаленої точки зображення.

Для отримання ГМО за допомогою цього виразу необхідні значні обчислювальні ресурси, що , в свою чергу, значно обмежує сферу застосування ГМО для систем автоматичного управління. Однак розрахунок ГМО можна значно скоротити, оскільки основна інформація про об’єкт міститься в його контурах і буде мати вигляд:

$$\mu'_p = \sum_{j=1}^{J} \Delta\rho \int_0^{2\pi} \rho^p I(\rho,\varphi) d\varphi \tag{3.3}$$

де $\Delta\rho = \dfrac{R_{\max}}{J}$, J – кількість точок контуру.

А для обчислення ГМО з використанням інформації виключно про ХТ звичайне інтегрування втрачає сенс. Тому для обчислення моментів пропонується використовувати інтегрування по Лебегу. Вираз для обчислення ГМО при цьому набуває вигляду [103]:

$$\mu_p = \iint_A \rho^p Q(\rho,\varphi) \psi(x \cap A), \tag{3.4}$$

де А – множина дійсних чисел; Х – підмножина дійсних чисел $x(\rho,\varphi)$ координат характерних точок контура фігури в полярній системі координат; ψ – міра на множині дійсних чисел; p – порядок момента; Q – множина ХТ контуру:

$$Q(\rho,\varphi)=\begin{cases}1, \rho=\rho_i, \varphi=\varphi_i; i=1,...,J.\\ 0, \rho\neq\rho_i.\end{cases} \quad (3.5)$$

де J – число ХТ контуру об’єкта.

Отже, в дискретному вигляді формула (3.4) буде мати вигляд:

$$\mu_p=\sum_{i=1}^{p}\rho_i^p\psi\left[x(\rho_i,\varphi_i)\cap A\right] \quad (3.6)$$

Врахувавши, що міра Лебега в полярній системі координат є площею трикутника $\Delta S=\frac{1}{2}\rho_{i+1}\rho_i\sin(\varphi_{i+1}-\varphi_i)$ можна записати формулу (3.6) в дискретному вигляді:

$$\mu_p=\left|\sum_{i=0}^{J}\rho_i^{*(p+2)}\sin(\Delta\varphi_i)\right|, \quad (3.7)$$

де $\rho_i^*=\sqrt{\rho_i\rho_{i+1}}$ – середнє геометричне радіус-векторів, $\Delta\varphi_i=\varphi_{i+1}-\varphi_i$ – різниця фаз між і-ою та (i+1)-ою ХТ.

Цей вираз береться по модулю, оскільки знак виразу залежить від напряму обходу характерних точок, а нас цікавить тільки абсолютне значення. Для забезпечення інваріантності до масштабу проведено нормування радіус-векторів використавши коефіцієнт $\rho=\sqrt{\frac{S}{2\pi}}$, де S – площа фігури.

Тоді формула (3.7) буде мати вигляд:

$$C_p=\left|\sum_{i=0}^{J}\rho_{i0}^{*(p+2)}\sin(\Delta\varphi_i)\right|, \quad (3.8)$$

де $\rho_{i0}^{*} = \dfrac{\rho_i^{*}}{\rho}$.

У випадку, коли $\Delta\varphi_i$ достатньо мале, тобто ХТ досить багато, можна вважати, що $\sin(\Delta\varphi_i) \approx \Delta\varphi_i$. Тоді отримаємо:

$$C_p = \left| \sum_{i=0}^{J} \rho_{i0}^{*(p+2)} \Delta\varphi_i \right|. \tag{3.9}$$

Значення C_p використовуються як ідентифікаційний вектор ознак.

Оскільки ХТ контуру голови не багато, то застосовуємо формулу (3.8) і для контуру із рисунку 2.17 для різних значень параметра a наведені приклади ідентифікаційних векторів у таблиці 3.1.

Таблиця 3.1 – Приклади ідентифікаційних векторів для контуру голови з рисунку 2.17

a =1	a =3	a =5	a =10
C1 = 18.1165415896	C1 = 18.1776031321	C1 = 18.2653010321	C1 = 18.6899241881
C2 = 26.3943845578	C2 = 26.5620245146	C2 = 26.7967078616	C2 = 27.9889287101
C3 = 38.8296080684	C3 = 39.1773192075	C3 = 39.6555130973	C3 = 42.1911847904
C4 = 57.6329614961	C4 = 58.2784847845	C4 = 59.1578268260	C4 = 63.9986873516
C5 = 86.2350323116	C5 = 87.3660901664	C5 = 88.9046863087	C5 = 97.6513576557
C6 = 129.9765046331	C6 = 131.8916626829	C6 = 134.5129442142	C6 = 149.8232588032
C7 = 197.1961129633	C7 = 200.3699819765	C7 = 204.7695178972	C7 = 231.0502016254
C8 = 300.9485303866	C8 = 306.1363758523	C8 = 313.4576181975	C8 = 358.0079482905
C9 = 461.7214762318	C9 = 470.1280293504	C9 = 482.2511539824	C9 = 557.1539062152
C10 = 711.7399301868	C10 = 725.2934234978	C10 = 745.3094482086	C10 = 870.5573502377

До переваг такого представлення відносять:

- Ознаки інваріантні до паралельного переносу, так як нуль полярної системи координат поєднується з центром ваги фігури;
- Ознаки стійкі до зміни масштабу, тому що радіус-вектори ХТ нормуються;

– Ознаки інваріантні до кута повороту об'єкта. Інваріантність забезпечується сумуванням по обходу контуру;

– Згладжувальні якості моментів визначають високу стійкість методу;

– Ознаки легко нарощуються до отримання стійкого розпізнавання;

– Мала кількість ХТ визначає високу обчислювальну ефективність методу.

3.2. Метод ієрархічної структурно-статистичної ідентифікації

В обробці зображень широко використовуються ієрархічні структури даних (піраміди) для моделювання і опису зображень [104]. Ієрархічна структура – це множина, яка упорядкована так, що існує тільки один елемент цієї множини, який не має попереднього, а всі інші елементи мають тільки один попередній. Це багаторівнева форма організації об'єктів з чіткою приналежністю об'єктів нижнього рівня певному об'єкту верхнього рівня.

Необхідність в багаторівневих уявленнях обумовлена появою завдань, що вимагають для свого рішення спрощення абстрактних описів і дискретних уявлень складних структур даних (зі збереженням важливих властивостей). Тому при проведенні операцій над об'єктами або областями зображення виникає проблема, яка полягає в розробці математичного апарату, який дозволяв би виконувати дії над частинами зображень як над елементарними об'єктами [87]. Ієрархічне уявлення усуває необхідність окремої обробки кожного елемента вихідного зображення, а саме: в той час як зазвичай кожна точка, що обробляється, має лише координати, положення в просторі, але не має форми. Пірамідальні структури дозволяють уявити елементарні такі частини зображень, які одночасно мають і розмір, і положення - це контури різних рівнів. Такий підхід дозволяє відносити до тієї чи іншої області цілі фрагменти зображення, відповідні елементам інтенсивності різних рівнів піраміди. Важливе також і взаємне розташування контурів, що визначають форму об'єкта на зображенні [25].

Пірамідальні (ієрархічні) структури використаємо для опису даних в задачах кластеризації і сегментації статичних зображень. Під ієрархічною сегментацією будемо розуміти послідовність розбиттів зображення на вкладені сегменти. Використання ієрархій контурів для представлення зображення дозволяє забезпечувати кращу апроксимацію об'єктів. З іншого боку, пірамідальне зображення, може бути сформовано на основі структури вхідного зображення [21].

Структурне представлення зображення за допомогою контурів полягає у представленні зображення у вигляді сукупності примітивних геометричних елементів (сукупності контурів) і їх відношення між собою, тобто таке представлення зображення побудоване на базі геометричного підходу, включаючи в опис форми взаємне положення і відношення окремих елементів. В такому випадку, зображення можна представити у вигляді сукупності зображень (контурів) різного рівня ієрархії:

$$F(x,y) = \sum_{i=1}^{n} I_i(x,y) \qquad (3.10)$$

де n – число рівнів ієрархії, $I_i(x,y)$ – зображення (контур) на i-му рівні ієрархії.

Суть методу ієрархічної структурно-статистичної ідентифікації полягає в застосуванні ієрархічної послідовності ідентифікаційних векторів, розрахованих на основі інформації про ХТ контурів об’єкта з регульованою деталізацією (рисунок 3.1).

В результаті проведення контурного аналізу зображення на базі запропонованого методу отримано пірамідальне представлення геометричних форм об’єкта і його деталей: набір контурних препаратів. Таке представлення використовується для обчислення геометричних ідентифікаційних ознак об’єкта. Даний підхід структурно-ієрархічної КС дозволяє підвищити швидкодію роботи системи відеоспостереження за рахунок зменшення об’єму інформації, необхідної для обробки [35].

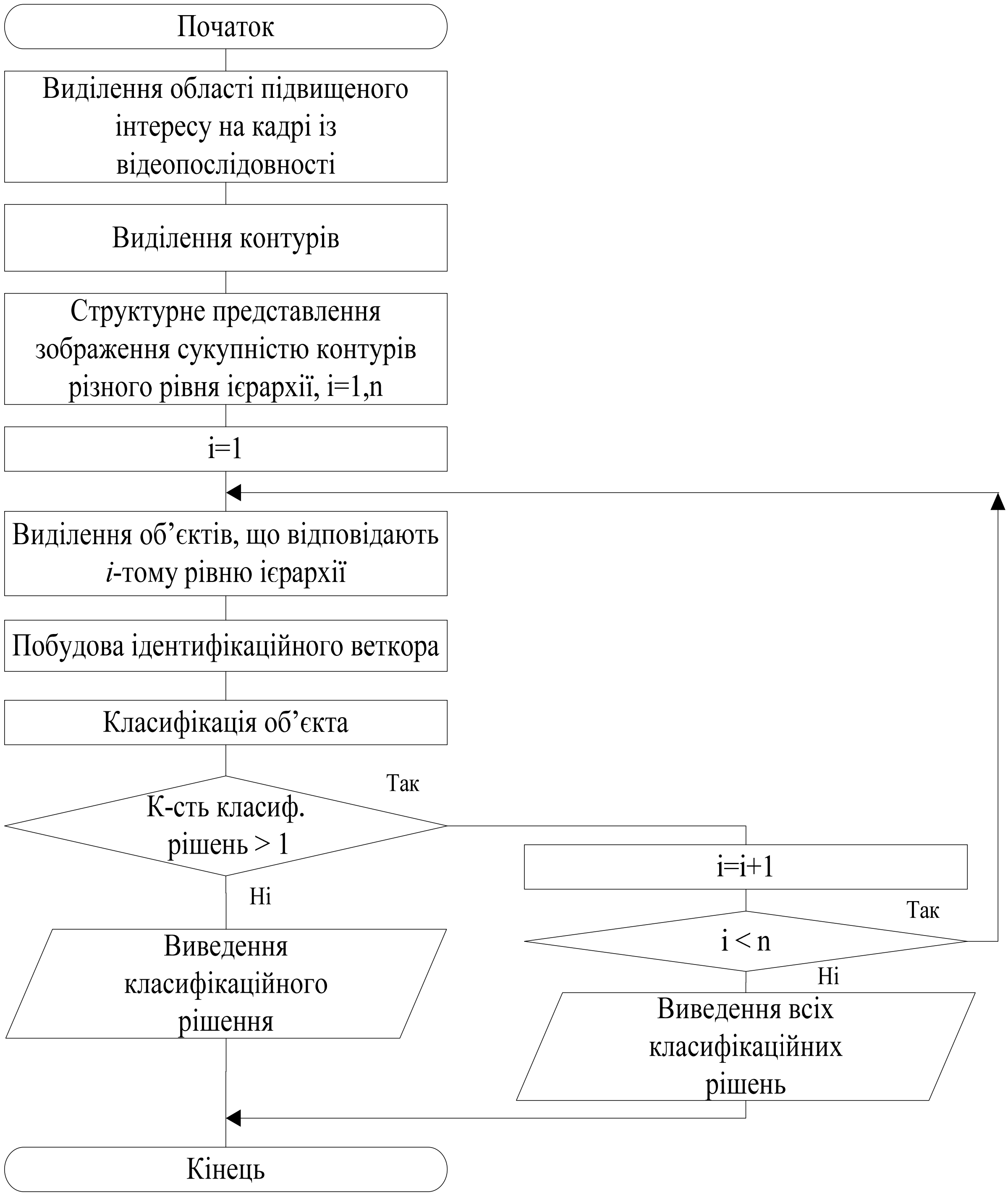


Рисунок 3.1 – Схема методу ієрархічної структурно-статистичної ідентифікації

Після того, як зображення представлено у вигляді сукупності контурів та їх відношень між собою з різним рівнем ієрархії, відбувається:

1) виділення ХТ контуру (спочатку на контурі найвищого рівня ієрархії);

2) побудова ідентифікаційного вектора на основі інформації про ХТ контуру;

3) проведення класифікації на відповідному рівні та віднесення об’єкта до відповідного класу;

4) перехід на нижчий рівень ієрархії та проводиться розпізнавання всередині класу, в який потрапив об’єкт (кроки 1-4). Ця процедура триває доти, поки не прийнято остаточного класифікаційного рішення або не пройдено всі рівні ієрархії.

Даний метод ієрархічної структурно-статистичної ідентифікації дозволяє підвищити швидкодію роботи АСВН за рахунок зменшення кількості об’єктів, з якими проводиться порівняння, на кожному рівні ієрархії.

3.3. Кластерний аналіз ідентифікаційних векторів

Задача класифікації (розбиття, кластеризації) добре вирішується методами кластерного аналізу, де множину об'єктів розташовують у такий спосіб, щоб усі вони належали одному класу (кластеру, групі), тобто були більш схожі один на одного, ніж на об'єкти інших кластерів [105].

Нехай $\mathbf{a}_{\mathrm{i,j}}$ - вектор ХТ *j*-того знімку *i*-тої особи, агрегований по всіх ступенях деталізації:

$$\mathbf{a}_{i,j} = \left(a_{i,j,1}, a_{i,j,2}, \ldots, a_{i,j,N}\right), \tag{3.11}$$

де $\mathrm{a}_{\mathrm{i,j,k}}$ - *k*-та ХТ *j*-того знімку *i*-тої особи, *N* - кількість ХТ. В свою чергу, $\mathrm{a}_{\mathrm{i,j,k}}$ - це також вектор, що описує декартові координати ХТ:

$$a_{i,j,k} = \left(ax_{i,j,k}, ay_{i,j,k}\right), \tag{3.12}$$

де $ax_{i,j,k}$ - абсциса *k*-тої ХТ *j*-того знімку *i*-тої особи, $ay_{i,j,k}$ - ордината *k*-тої ХТ *j*-того знімку *i*-тої особи.

Якщо дослідження вимагає використання не усіх рівнів деталізації, а якоїсь їх частини, то використовується відповідна підмножина з агрегованої множини ХТ $\mathbf{a}_{i,j} = \left(a_{i,j,1}, a_{i,j,2}, ..., a_{i,j,N}\right)$.

Для забезпечення інваріантності процедури класифікації до афінних перетворень зміни масштабу та зсуву, доцільно пронормувати ХТ відносно їхнього центроїда та геометричних розмірів об‘єкта:

$$na_{i,j,k} = s_{i,j}\left(ax_{i,j,k} - cx_{i,j}, ay_{i,j,k} - cy_{i,j}\right), \tag{3.13}$$

де $\mathrm{na}_{\mathrm{i,j,k}}$ - *k*-та нормована ХТ *j*-того знімку *i*-тої особи, $\mathrm{s}_{\mathrm{i,j}}$ - масштабуючий коефіцієнт *j*-того знімку *i*-тої особи, $\mathrm{cx}_{\mathrm{i,j}}$ - абсциса центроїду *j*-того знімку *i*-тої особи, $\mathrm{cy}_{\mathrm{i,j}}$ - ордината центроїду *j*-того знімку *i*-тої особи.

Масштабуючий коефіцієнт $s_{i,j}$ та координати центроїда $\left(cx_{i,j}, cy_{i,j}\right)$ можна оцінити таким чином:

$$s_{i,j} = \frac{1}{\left(\max\limits_{k} ax_{i,j,k} - \min\limits_{k} ax_{i,j,k}\right)}, \tag{3.14}$$

$$cx_{i,j} = \frac{1}{N}\sum_{k=1}^{N} ax_{i,j,k}, \tag{3.15}$$

$$cy_{i,j} = \frac{1}{N}\sum_{k=1}^{N} ay_{i,j,k}. \tag{3.16}$$

Нехай $\mathbf{b}$ - вектор ХТ нового знімку:

$$\mathbf{b} = \left(b_1, b_2, ..., b_N\right). \tag{3.17}$$

Для класифікації вектора ХТ нового знімку $\mathbf{b}$ можна використати кластерний аналіз. При цьому, кожна особа описується множиною ХТ усіх її знімків, які являють собою один кластер:

$$A_i = \left(\mathbf{a}_{i,1}, \mathbf{a}_{i,2}, ...\right), \tag{3.18}$$

де $\mathrm{A_i}$ - множина ХТ *i*-того кластера.

Відстань між двома довільними векторами ХТ $\mathbf{a} = \left(a_1, a_2, ..., a_N\right)$ та $\mathbf{b} = \left(\mathrm{b}_1, \mathrm{b}_2, ..., \mathrm{b}_\mathrm{N}\right)$ можна оцінити як евклідову норму

$$\rho_1(\mathbf{a}, \mathbf{b}) = \left\|\mathbf{a} - \mathbf{b}\right\|_1 = \sqrt{\sum_{k=1}^{N} \left(ax_k - bx_k\right)^2 + \left(ay_k - by_k\right)^2}, \tag{3.19}$$

де ax_k, ay_k - абсциса та ордината k-тої ХТ вектора $\mathbf{a}$, bx_k, by_k - абсциса та ордината k-тої ХТ вектора $\mathbf{b}$.

Альтернативно, для забезпечення інваріантності процедур класифікації до афінних перетворень повороту, можна скористатися формулою модифікованої евклідової відстані. Вона описує відмінність відстаней окремо-взятих характерних точок від початку координат:

$$\rho_2(\mathbf{a}, \mathbf{b}) = \left\|\mathbf{a} - \mathbf{b}\right\|_2 = \sqrt{\sum_{k=1}^{N} \left(ax_k^2 + ay_k^2 - bx_k^2 - by_k^2\right)^2}. \tag{3.20}$$

Класифікацію ХТ нового знімку $\mathbf{b}$ можна проводити в різний спосіб. Ці способи відрізняються методом обчислення міри належності, іншими словами – подібності, $\mathrm{similarity}(\mathbf{b}, \mathrm{A_i})$ вектора $\mathbf{b}$ до кластера A_i:

$$0 < similarity(\mathbf{b}, A_i) \le 1. \tag{3.21}$$

Що більше значення функції подібності $similarity(\mathbf{b}, A_i)$, то вища ймовірність, що ХТ зображення **b** належить до кластера A_i. Коли значення функції подібності перевищує деякий поріг, то можна вважати, що знімок **b** належить до відповідного кластера.

Розглянемо альтернативні підходи до обчислення $similarity(\mathbf{b}, A_i)$. Існують наступні методи кластерного аналізу без попереднього навчання [106]:

Ієрархічні методи:

- метод ближнього зв'язку;
- метод середнього зв'язку Кінга;
- метод Уорда.

Ітеративні методи угруповання:

- метод k-середніх Мак-Куіна.

Алгоритми типу розрізання графа:

- метод кореляційних плеяд Терентьєва:
- вроцлавська таксономія.

Зважаючи на специфіку задачі додавання нового елемента до одного з існуючих кластерів, доцільно використати метод k-середніх, де значення центроїдів у кластерах апріорі відоме.

Метод k-середніх Мак-Куіна [107] - це метод кластерного аналізу, метою якого є поділ m спостережень (об'єктів) на k кластерів, при цьому кожне спостереження відноситься до того кластеру, до центру (центроїду) якого воно є найближче.

Функція подібності за методом k-середніх з фіксованими центрами обчислюється наступним чином:

$$similarity_1(\mathbf{b}, A_i) = \frac{1}{1+\rho\left(\mathbf{b}, \mu(A_i)\right)}, \qquad (3.22)$$

де $\rho(\mathbf{b}, \mu(A_i))$ – функція відстані між векторами характерних точок, що обчислюється за формулою (3.19) або (3.20), $\mu(A_i)$ – центроїд кластера A_i, що обчислюється за формулою

$$\mu(A_i) = \frac{1}{|A_i|} \sum_{\mathbf{a} \in A_i} \mathbf{a}, \tag{3.23}$$

де $|A_i|$ – потужність *i*-того кластера, тобто кількість векторів ХТ, що належать до цього кластера.

Другим підходом є обчислення функції подібності через суму відстаней вектора ХТ $\mathbf{b}$ до усіх векторів кластера A_i:

$$similarity_2(\mathbf{b}, A_i) = \frac{1}{1 + \sum_{\mathbf{a} \in A_i} \rho(\mathbf{a}, \mathbf{b})}, \tag{3.24}$$

де $\rho(\mathbf{a}, \mathbf{b})$ – функція відстані між векторами характерних точок, що обчислюється за формулою (3.19) або (3.20).

Третім і останнім підходом буде обчислення функції належності як максимум подібності по всіх векторах кластера A_i:

$$similarity_3(\mathbf{b}, A_i) = \max_{\mathbf{a} \in A_i} \frac{1}{1 + \rho(\mathbf{a}, \mathbf{b})}. \tag{3.25}$$

Після занесення ХТ нової відеоінформації до того чи іного кластеру, повинна виконуватися умова мінімізації сумарного квадратичного відхилення точок кластера від центроїдів даних кластерів:

$$\min\left[\sum_{i=1}^{k} \sum_{\mathbf{a} \in A_i} \|\mathbf{a} - \boldsymbol{\mu}(A_i)\|^2\right], \tag{3.26}$$

де $\|\mathbf{a}-\boldsymbol{\mu}(A_i)\|$ - відстань між векторами характерних точок, обчислена за однією з формул (3.19) або (3.20), $\boldsymbol{\mu}(A_i)$ - центроїд для кластеру A_i.

Використання описаного методу викликає труднощі в узгодженні результатів між різними рівнями деталізації. З метою подолання цих труднощів доцільно використовувати алгоритм паритетного голосування.

Суть алгоритму полягає у знаходженні одного голового елементу серед більшості. Головним є той що повторюється більше половини разів за наявну кількість едементів. Якщо немає більшості, алгоритм не виявить цього факту, і все одно виведе один із елементів.

Алгоритм може взагалі не знайти елемента, який має найбільше повторення, якщо кількість повторів не є достатньо великою.

В четвертому розділі проведено дослідження ефективності описаних тут методів класифікації.

Висновки до розділу 3

У даному розділі отримав подальший розвиток метод структурно-статистичної ідентифікації об'єктів на базі застосування ієрархічної послідовності ідентифікаційних векторів на основі інформації про координати ХТ контуру відповідного рівня ієрархії.

Запропоновано три підходи до класифікації об'єктів: на основі центроїда, суми і максимуму та дві альтернативні схеми для обчислення відстані між векторами ХТ: класичною евклідовою метрикою та модифікованою на основі відстаней від початку координат.

Структурне ієрархічне представлення зображення грунтується на основі геометричного підходу і включає опис форми і взаємне розміщення та відношення окремих елементів. Для контуру *i*-го рівня ієрархії проводиться виділення ХТ, а на їх основі будується ідентифікаційний вектор на базі ГМО. І на основі ідентифікаційного вектора проводиться класифікація.

На кожному рівні ієрархії об’єкт відноситься до одного із класів. А на наступному рівні ієрархії проводиться порівняння тільки із тими об’єктами, що відносяться до цього рівня і т.д. поки не буде прийнято класифікаційне рішення або не буде нижчого рівня ієрархії.

До переваг використання ГМО відносяться: інваріантність до паралельного перенесення та повороту об'єкта за рахунок сумування по обходу контуру, незалежність від масштабу, стійкість, а мала кількість ХТ визначає обчислювальну ефективність.

За рахунок використання структурно-ієрархічної контурної сегментації дозволяє підвищити швидкодію роботи АСВН за рахунок зменшення об’єму інформації необхідної для обробки на кожному рівні ієрархії на декілька порядків.

Розроблений підхід дав змогу успішно ідентифікувати контури зображень.

Експериментальні дослідження розроблених у цьому розділі методів наведені у розділі 4.

Основні результати розділу опубліковані в працях [31, 35, 40, 47].

РОЗДІЛ 4
ПРИКЛАДНА ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ РОЗПІЗНАВАННЯ ОБ'ЄКТІВ НА ОСНОВІ РОЗРОБЛЕНИХ МЕТОДІВ

4.1. Структура інформаційної технології

Для ефективної організації процесу розпізнавання об’єктів на основі розроблених методів було створено інформаційну технологію, що дозволяє впорядковувати і обробляти відеоінформацію, виявляти на ній характерні ознаки та проводити на цій основі класифікацію нових зразків відеоінформації. На рисунку 4.1 представлена структура АСВН.

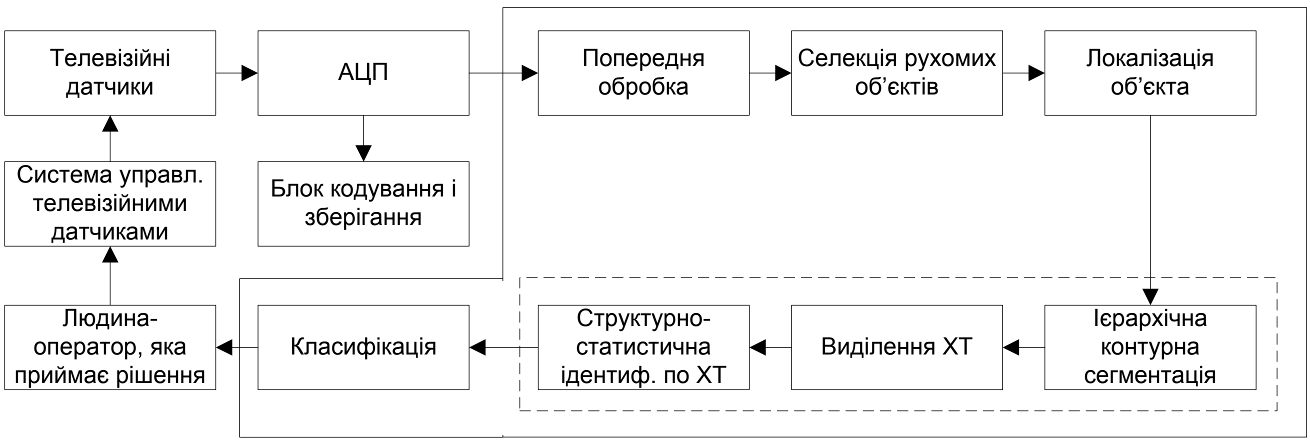


Рисунок 4.1 – Структура АСВН із пропонованими підсистемами

Запропонована АСВН складається із:

1) Телевізійних датчиків (фото, відеокамери), які забезпечують ефективний механізм створення електронної версії зображення;

2) АЦП (аналого-цифрового перетворювача) – здійснює перетворення та стиснення зображень у підтримувані формати, а блок кодування і зберігання – здійснює запис та відтворення цифрових відеосигналів;

3) Попередня обробка зображення, яка полягає у створенні умов, що підвищують ефективність і якість виділення і розпізнавання шуканих або досліджуваних об'єктів;

4) Селекція рухомих об’єктів – дозволяє автоматично оцінювати зміну візуальної обстановки на об'єктах, що переглядаються телевізійними камерами, використовується для оповіщення про появу рухомих об'єктів в охоронній зоні, базується на порівняльному аналізі зображень «поточного» і «попереднього» кадрів, занесених в кадрову пам'ять;

5) Після селекції об’єкту відбувається його локалізація – в нашому випадку виділяється область обличчя на кадрі із відео потоку;

6) Ієрархічна контурна сегментація – полягає у розбитті зображення на вкладені контури. Результатом даної сегментації зображення є множина контурів, виділених із зображення, що дає змогу значно зменшити об’єм інформації що обробляється та підвищує швидкодію всієї АСВН;

7) На контурі кожного рівня ієрархії проводимо виділення ХТ (координати ХТ заносимо у масив);

8) Після чого проводимо структурно-статистична ідентифікацію об’єктів – обчислюємо C_p на основі методу ГМО;

9) Проводимо класифікацію – процедура автоматичного віднесення досліджуваного об’єкту (зображення обличчя), заданого сукупністю спостережень, до одного із взаємовиключних класів людей, або формування висновку про те, що об’єкт не відноситься до жодного із відомих класів.

10) На основі класифікаційного рішення людина-оператор приймає остаточне рішення (навести камеру ближче, повідомити службу охорони, проігнорувати).

Розглянемо детальніше ієрархічну КС, виділення ХТ контуру та структурно-статистична ідентифікацію об’єктів.

Застосувавши метод КС на базі методу Канні, на першому рівні ієрархії виділяємо контур обличчя. Після цього здійснюємо процедуру прослідковування контуру та здійснюємо виділення ХТ контуру методом описаним у розділі 2 (рисунок 4.2). На основі інформації про ХТ будуємо ідентифікаційний вектор та проводимо класифікацію, результатом якої є віднесення контру голови (людини) до одного із семи класів. Використано наступний поділ облич на класи: овальне, кругле, квадратне, прямокутне, трикутне, грушоподібне, ромбоподібне .

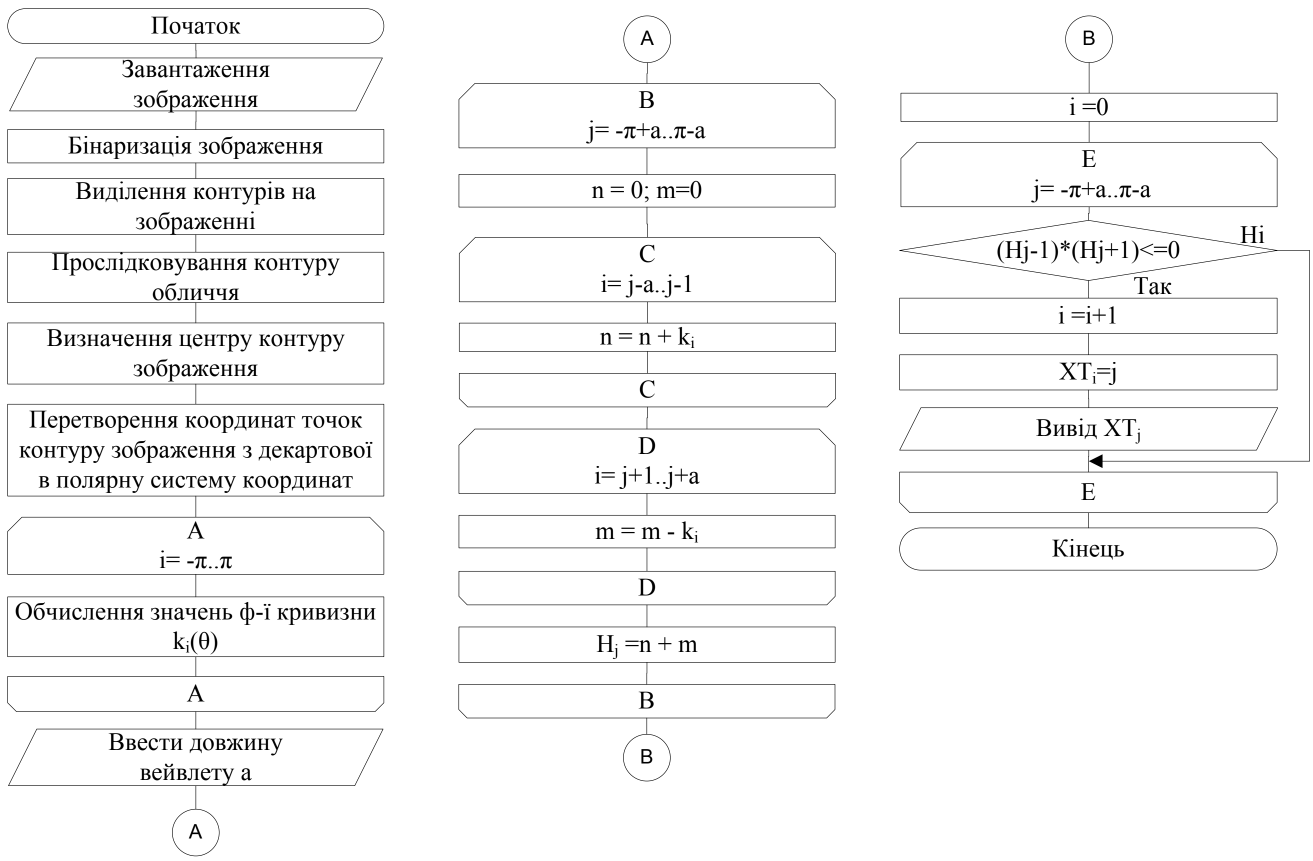


Рисунок 4.2 – Схема алгоритму роботи модуля виявлення ХТ

Якщо потрібно знайти конкретну людину в потоці людей, то вже на цьому (першому) рівні ієрархії значна частина людей відсіюється і обробка їхнього зображення не має змісту.

Якщо цієї інформації недостатньо (людина залишається підозрюваною), то переходимо на другий рівень ієрархії, де зменшуємо довжину вейвлета та працюємо не із всім зображенням, а тільки з тою його частиною, що знаходиться в середині контуру вищого рівня ієрархії (в середині контуру голови) та повторюємо дії: виділяємо контур носа, знаходимо ХТ контуру носа та проводимо поділ облич на класи за формою носа. Тут виділили наступні чотири класи:

вузький високий ніс, середній, низький, широкий та дуже широкий. Якщо класифікаційне рішення не прийняте, то переходимо на наступні рівні ієрархії та проводимо ідентифікацію за контуром рота та очей.

Як показано на рисунку 4.1, АСВН оперує як поточними вхідними так і збереженими та внесеними для пошуку даними. Такий великий набір даних, зазвичай зберігають в базі даних для забезпечення взаємопов'язаності, мінімальної надлишковості, незалежності від програм, цілісності та захисту від неавторизованого доступу [108, 109].

Предметну область підсистеми структурно ієрархічної ідентифікації АСВН можна представити в табличному вигляді [110-112] тому, доцільно використати реляційну модель бази даних, з підтримкою систем управління базами даних [109], зокрема SQLite [113], котра забезпечить максимальну швидкодію та ефективність роботи з даними.

Для проектування інфологічної моделі предметної області підсистеми ієрархічної ідентифікації використано графічну мову ER-моделювання [114] (рисунок 4.3).

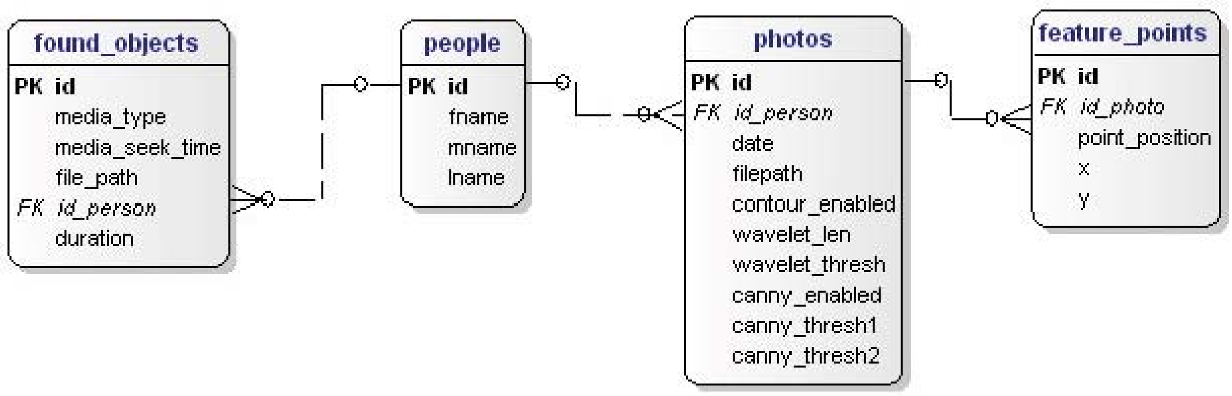


Рисунок 4.3 – Схема даних підсистеми структурно ієрархічної ідентифікації

Структура таблиць БД представлена в додатку Г. У таблиці «people» описано інформацію про внесені в систему об'єкти, опис моделі даних у форматі (data definition language, DDL):

```
-- Table: people
CREATE TABLE people (
  id INTEGER PRIMARY KEY AUTOINCREMENT UNIQUE,
  fname VARCHAR (100),
  mname VARCHAR (100),
  lname VARCHAR
);
```

В таблиці «photos» містяться файли зображень об’єктів для пошуку, параметри налаштування довжини і розмір порогу вейвлет-функцій, розміри порогів запропонованого методу на базі методу Канні, опис моделі даних у форматі DDL:

```
-- Table: photos
CREATE TABLE photos (
  id INTEGER PRIMARY KEY AUTOINCREMENT UNIQUE,
  id_person INTEGER REFERENCES people (id) ON DELETE CASCADE,
  date DATETIME,
  filepath VARCHAR (255),
  contour_enabled BOOLEAN,
  wavelet_len INTEGER,
  wavelet_thresh DOUBLE,
  canny_enabled BOOLEAN,
  canny_thresh1 DOUBLE,
  canny_thresh2 DOUBLE
);
```

В таблиці «feature_points» міститься інформація про ХТ контурів зображень об’єктів для пошуку, опис моделі даних у форматі DDL:

```
-- Table: feature_points
CREATE TABLE feature_points (
  id INTEGER PRIMARY KEY AUTOINCREMENT UNIQUE,
  id_photo INTEGER REFERENCES photos (id) ON DELETE CASCADE,
  point_position INTEGER,
```

```
  x DOUBLE,
  y DOUBLE
);
```

В таблиці «found_objects» міститься інформація про знайдені об’єкти: типи файлів що оброблялись, час та дату знайдених у них об’єктів з таблиці «people», , опис моделі даних у форматі DDL:

```
-- Table: found_objects
CREATE TABLE found_objects (
  id INTEGER PRIMARY KEY AUTOINCREMENT UNIQUE,
  media_type VARCHAR (100),
  media_seek_time BIGINT,
  file_path VARCHAR (500),
  id_person INTEGER REFERENCES people (id) ON DELETE CASCADE,
  duration BIGINT
);
```

Отже, на базі СУБД SQLite створено БД, що забезпечило гнучкий доступ та управління даними під час роботи АСВН.

4.2. Об’єктно-орієнтоване моделювання і програмування

Програмне забезпечення (ПЗ) реалізовано з використанням об’єктно-орієнтованого підходу, зразків проектування (design patterns), та методів мінімізації складності компонентів. В об’єктно-орієнтованій програмі сутності предметної області представляються абстрактними компонентами (класами). Рівень абстрагування визначається специфікою задачі що дозволяє уникнути несуттєвих для розв’язання задачі деталей.

Зразки проектування являють собою стандартні розв’язки типових задач програмування. При розробці ПЗ використано кілька зразків проектування, зокрема: singleton, dependency injection, model-view-presenter [115].

Зразок проектування singleton дозволяє організувати доступ до компонента з різних місць програми в зручний та ефективний спосіб. У програмі АСВН singleton використано при реалізації об'єктів доступу до даних, оскільки вони використовуються з численних місць програми і ефективність їхнього використання впливає на загальну ефективність функціонування програми.

Зразок проектування dependency injection призначений для відокремлення компонентів з метою їх пізнього зв'язування. Пізнє зв'язування компонентів дозволяє підвищити гнучкість програми, її готовність до змін та створення компонентів для повторного використання [116]. Цей зразок проектування слугує основою для наступного зразка проектування - model-view-presenter.

Зразок проектування model-view-presenter дозволяє відокремити логіку опрацювання даних від їхнього відображення. . У свою чергу, це дає можливість, за допомогою технологій імітування залежностей (mocking) розробляти вичерпні автоматизовані тести та проектувати крос-платформне програмне забезпечення.

З метою мінімізації складності компонентів використано сучасні підходи програмування на основі лямбда-виразів та бібліотек ланцюжкового опрацювання колекцій. Лямбда-вирази дозволяють у стислій формі описувати предикати для фільтрування множини об'єктів та функції перетворення множини елементів.

Бібліотеки ланцюжкового опрацювання колекцій дозволяють записувати складні послідовності обробки колекції даних у лаконічному вигляді послідовних викликів методів де результат попереднього виклику передається як вхід для наступного і т.д.

Описані підходи застосовані при реалізації таких груп компонентів ПЗ: моделей даних, об'єктів доступу до даних, класів MVP для внесення нових об'єктів та компонентів MVP для виділення контурів у файлах відеоінформації.

Моделі даних програмної реалізації сутностей бази даних АСВН описуються наступними компонентами:

- Person,
- Photo,
- FeaturePoint,

– DetectedPerson.

UML – діаграма їхніх класів наведена на рисунку рисунок 4.4 [117]. Модель «Person» описує особу котру виявлятиме АСВН. Ця модель містить усі властивості відповідної таблиці БД та додаткові методи що дозволяють виводити інформацію про особу у читабельній формі та створювати нові об’єкти цієї моделі.

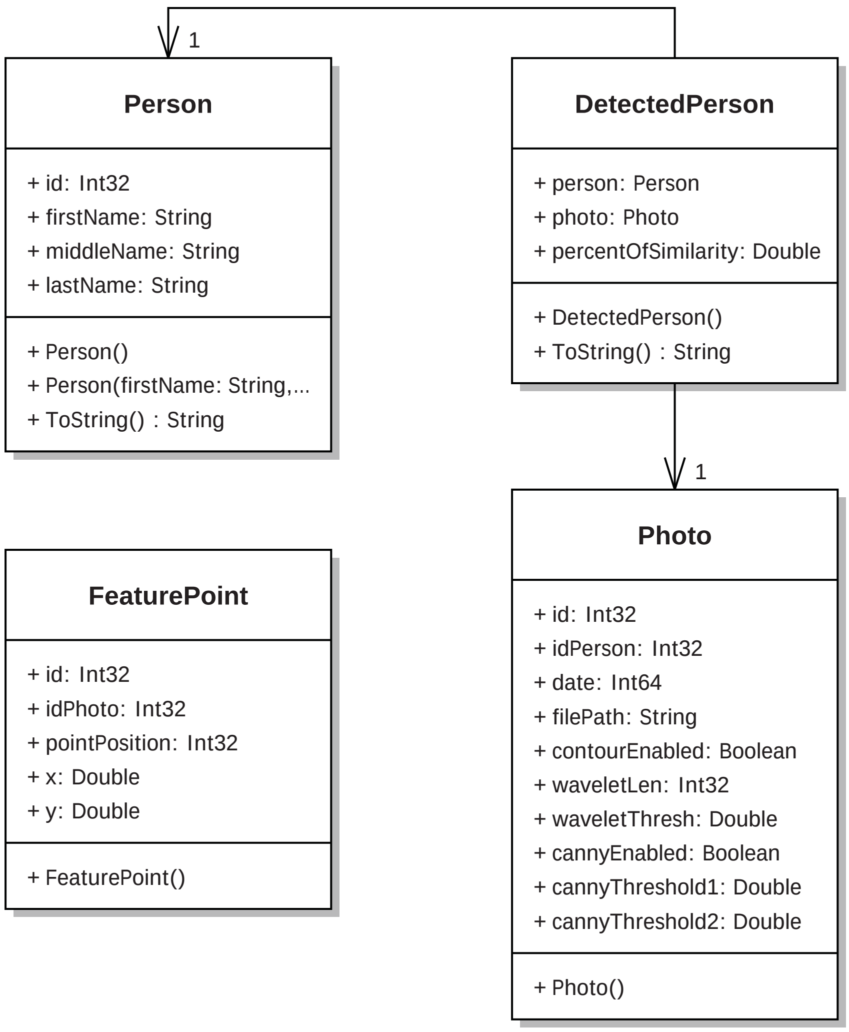


Рисунок 4.4 – UML – діаграма моделей даних АСВН

Модель «Photo» містить налаштування параметрів методів контурної сегментації, шлях до файлу відеоінформації виявленого обличчя і дату внесення в БД. Модель «FeaturePoint» виступає відображенням таблиці ХТ. Модель «DetectedPerson» описує міру подібності множини ХТ нового зображення до тієї чи іншої особи. Ця модель пов’язана з моделлю «Person» та «Photo» таким чином, що поле *percentOfSimilarity* описує процент подібності ХТ до конкретного знімку *photo* особи *person*.

Доступ до даних прийнято організовувати за допомогою так званих об'єктів доступу до даних (Data Access Objects, DAO). Така організація дозволяє легко переходити від однієї СУБД до іншої та спрощує процес модульного (Unit) тестування програмного забезпечення. Об'єкти доступу до даних АСВН зображені на рисунку 4.5.

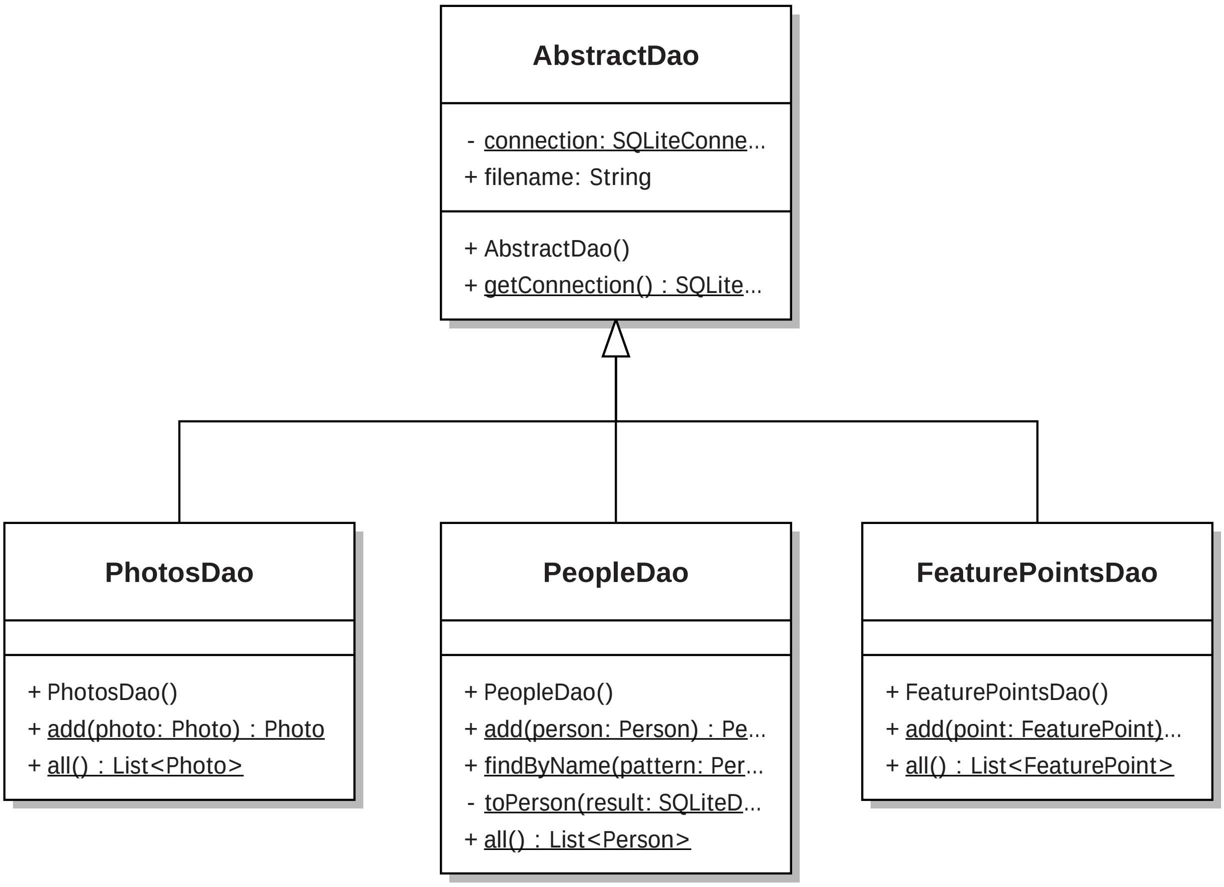


Рисунок 4.5 – UML – діаграма класів об'єктів доступу до даних

В компоненті «AbstractDao» зібрано спільні властивості усіх об'єктів доступу до даних, а саме: метод getConnection, який здійснює відкладену (Lazy) ініціалізацію поля «connection» за допомогою інкапсульованої стрічки з'єднання з БД. Значення цієї стрічки дозволяє змінювати тип СУБД. Оскільки в даному дослідженні використовується СУБД SQLite, що зберігає дані у файловій системі, тому поле «filename» задає шлях до БД.

Компонент «PeopleDao» дозволяє виконувати операції створення, пошуку та отримання колекцій усіх об’єктів класу «Person». Для створення нового об’єкту використовується метод «add». Цей метод отримує на вхід зразок об’єкту, що описує нову особу, а на виході повертає цей же об’єкт, доповнюючи його унікальним ідентифікатором (id) новоствореного запису у БД.

Метод «findByName» виконує пошук об’єкта особи за зразком його прізвища, імені та по-батькові. Результатом виконання методу є об’єкт особи з унікальним ідентифікатором, у випадку присутності відповідного запису у БД, інакше метод поверне посилання на порожній об’єкт.

Результатом виконання методу «all» є колекція об’єктів всіх осіб, що були внесені у БД. Причому, ця колекція може бути порожньою.

Приватний метод «toPerson» призначений для перетворення нетипізованої сутності БД у строго типізований об’єкт «person».

Компоненти «PhotosDao» та «FeaturePointsDao» виконують схожу до «PeopleDao» функцію, але над об’єктами класів «Photo» та «FeaturePoint». Тобто їхні методи «add» створюють об’єкт у БД та доповнюють його унікальним ідентифікатором, а методи «all» виконують зчитування цих даних.

Для програмної реалізації обрано парадигму model-view-presenter (MVP), що дозволяє мінімізувати складність програмного забезпечення та полегшити написання модульних тестів. Уся логіка опрацювання даних, яку прийнято називати бізнес-логікою, зібрана у презентаційних компонентах (presenter). Відображення даних у інтерфейсі користувача (UI) виконується відображувальними компонентами, що реалізують абстракцію вигляду (view). Дані, якими обмінюються презентаційні та відображувальні компоненти називаються моделями (model).

На рисунку 4.6 наведено діаграму класів MVP для внесення нових об’єктів в АСВН. Компонент «AddFaceView» забезпечує опосередкований доступ до множини параметрів методів виділення контурів, що отримуються із інтерфейсу користувача, зокрема реалізованого компонентом «AddFaceForm». Опосередкованість доступу дозволяє легко та безпомилково добавляти нові види користувацького інтерфейсу без жодних змін компоненту опрацювання відеоданих «AddFacePresenter».

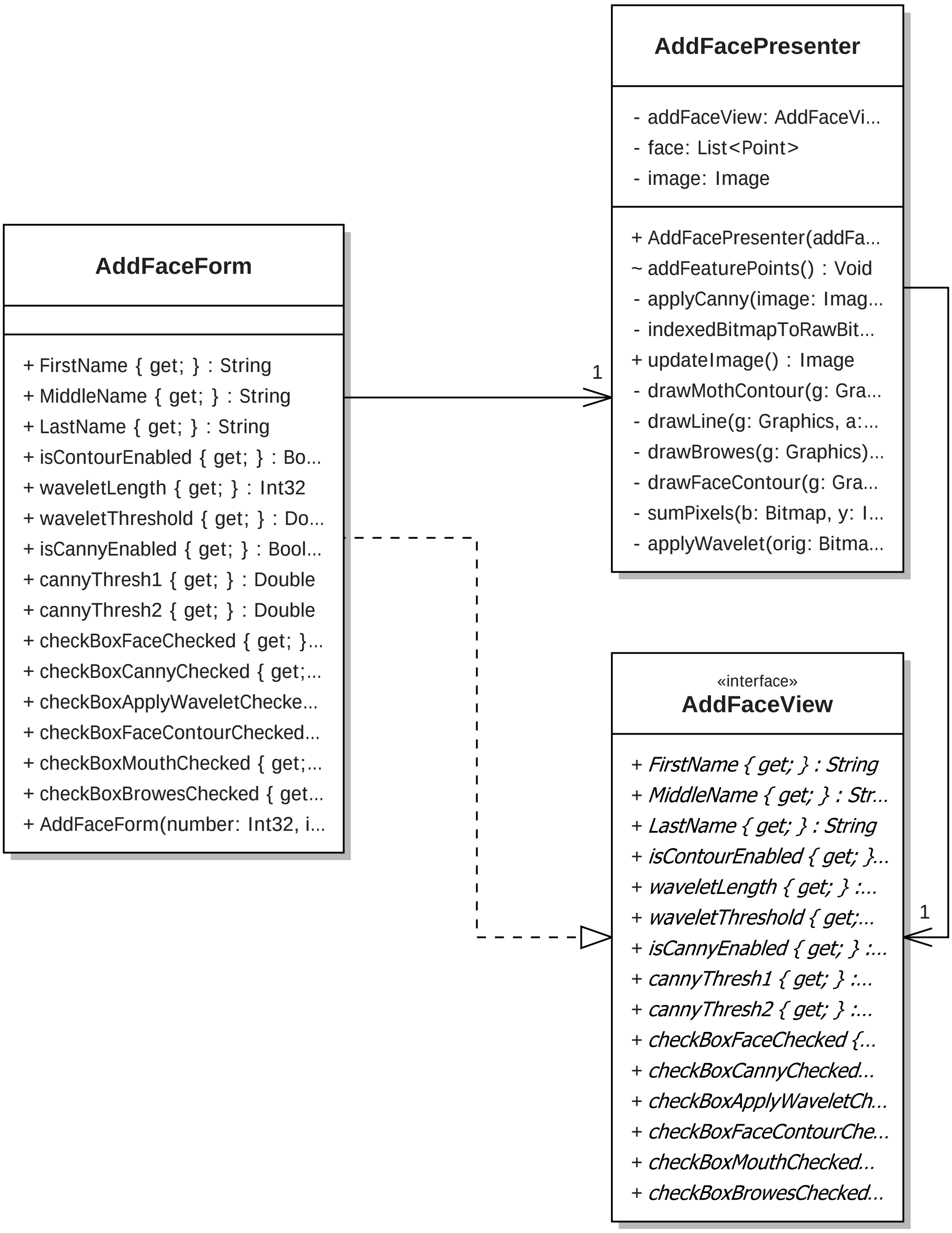


Рисунок 4.6 – Діаграма класів MVP для внесення нових об’єктів

Компонент «AddFacePresenter» вміщує методи для відображення контурів відеоінформації на різних рівнях деталізації із застосуванням різних значень порогів, згідно з налаштуваннями компоненту «AddFaceView».

Метод «applyCanny» виконує фільтрування вхідної відеоінформації за методом Канні, при цьому задіюючи параметри «isCannyEnabled», «cannyThresh1» та «cannyThresh2». Перед обробкою він зберігає копію початкової відеоінформації, що дозволяє легко експериментувати зі значеннями параметрів і відміняти попередні перетворення.

Метод «applyWavelet» виконує вейвлет-перетворення відеоінформації із застовуванням параметрів «checkBoxApplyWaveletChecked», довжини вейвлета «waveletLength» та вейвлет-порогу «waveletThreshold». Для своєї роботи цей метод використовує допоміжну функцію «sumPixels», що виконує додавання значень яскравості суміжних пікселів у рядку відеоінформації.

Метод «updateImage», на основі двох попередньо згаданих методів, формує результуюче зображення та дозволяє накладати на нього характерні ознаки, виявлені на різних ступенях деталізації з використанням допоміжних функцій «drawFaceContour», «drawMounthContour» і «drawBrowes». Ці функції, у свою чергу, використовують процедуру «drawLine», що зображує графічні примітиви на результуючому зразку відеоінформації.

На рисунку 4.7 наведено діаграму компонентів MVP для виділення контурів у файлах відеоінформації. Компонент «FeaturePointComparationType» описує можливі методи обчисленні відстані між векторами ХТ згідно формул (3.19), (3.20). Відповідний метод можна обрати за допомогою пункту меню «Налаштування» в основному вікні АСВН.

Компонент «FacesService» виконує обчислення міри подібності вектора ХТ нового зразка відеоінформації до наявних у базі кластерів ХТ відомих облич. Обчислення виконуються за допомогою формул (3.19) – (3.25).

Компонент «ImageUtils» виконує нормування векторів ХТ згідно з формулами (3.13) – (3.16).

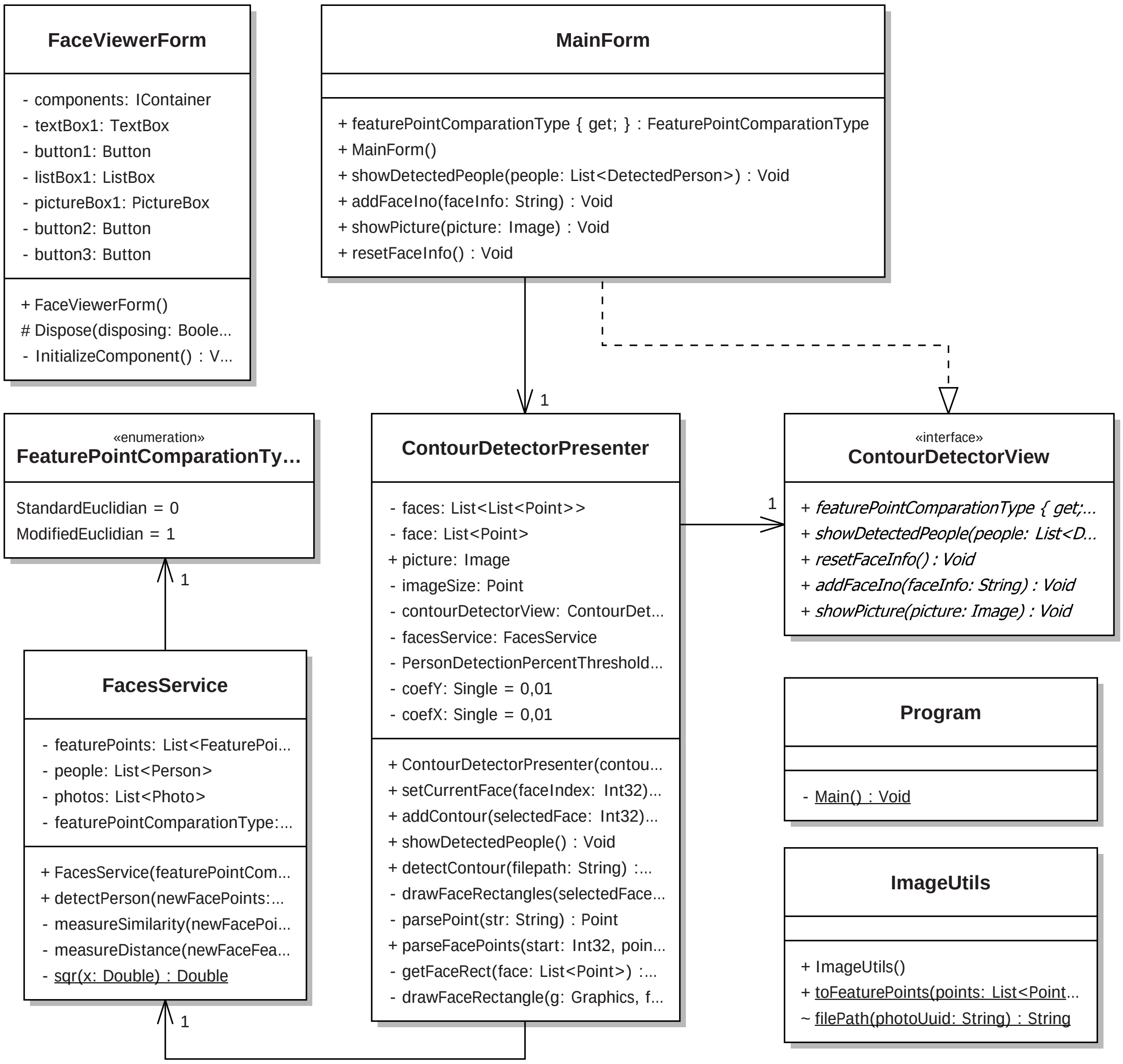


Рисунок 4.7 – Діаграма компонентів MVP для виділення контурів у файлах відеоінформації

Компоненти «ContourDetectorPresenter», «ContourDetectorView», «MainForm» призначені для виділення множини облич у файлі відеоінформації та забезпечують інтерактивний перегляд результатів їхньої класифікації за допомогою кластерного аналізу.

Метод «detectContour» компоненту «ContourDetectorPresenter» отримує на вхід шлях до файлу із відеоінформацією і користуючись допоміжною функцією «parseFacePoints» формує список ХТ для усіх виявлених на відеоінформації об'єктів облич. Функція «parseFacePoints» реалізована у рекурсивному вигляді, обробляючи окремі ХТ функцією «parseFacePoint».

Метод «drawFaceRectangles» з допомогою функцій «drawFaceRectangle» та «getFaceRect» візуально виокремлює виявлені у відеоінформації об'єкти облич, виділяючи їх прямокутною рамкою з порядковим номером.

Метод «showDetectedPeople» ініціює процес відображення результатів кластерного аналізу щодо класифікації виявлених у відеоінформації об'єктів облич. Він викликається при кожній зміні типу методів класифікації та їхніх параметрів.

Компонент «Program» виконує запуск головного вікна АСВН та підготовляє необхідні для її роботи ресурси.

Компонент «FaceViewerForm» призначений для перегляду, реагування та вилучення векторів ХТ виявлених облич у БД.

Описані компоненти були реалізовані як ІТ АСВН ідентифікації об'єктів в середовищі Visual Studio 2015 для платформи .NET на мові програмування C# використанням бібліотек OpenCV [58]. Дана розробка була виконана в рамках проекту № 0112U008453. Основні модулі ІТ АСВН дозволяють завантажувати та здійснити аналіз відеозображень, виділяти контури облич та ХТ контурів, будувати ідентифікаційні вектори, на основі яких здійснюється класифікація об'єктів.

Формули (3.19) та (3.20) реалізовані на мові C# наступним чином:

```
private double measureDistance(List<FeaturePoint> newFaceFeaturePoints,
List<FeaturePoint> photoFeaturePoints)
  {
   if (photoFeaturePoints.Count == 0)
   {
    return Double.MaxValue;
   }
   var N = Math.Min(photoFeaturePoints.Count, newFaceFeaturePoints.Count);
   var sumDiff = 0.0;
```

```
    for (var i = 0; i < N; ++i)
    {
     if (featurePointComparationType ==
FeaturePointComparationType.StandardEuclidian)
     {
      sumDiff += sqr(photoFeaturePoints[i].x - newFaceFeaturePoints[i].x) +
       sqr(photoFeaturePoints[i].y - newFaceFeaturePoints[i].y);
     }
     else
     {
      sumDiff += sqr(sqr(photoFeaturePoints[i].x) +
sqr(photoFeaturePoints[i].y) -
       sqr(newFaceFeaturePoints[i].x) - sqr(newFaceFeaturePoints[i].y));
     }
    }
    return Math.Sqrt(sumDiff);
   }
```

На початку роботи метод «measureDistance» перевіряє чи список ХТ є непорожнім. Якщо він порожній, то відстань між векторами ХТ оцінюється як $+\infty$. Це дає змогу уникнути хибної класифікації об'єктів на яких ХТ невиявлені. Для обчислення відстані виконується обхід списку ХТ і для кожної з них, відповідно до значення параметру «featurePointComparationType» обчислюється або стандартна евклідова відстань (3.19) або модифікована (3.20). Функція «sqr» виконує піднесення дійсного числа до квадрату.

Формула (3.22) класифікації об'єктів методом центроїда реалізована на мові C# наступним чином:

```
private DetectedPerson measureSimilarityCentroid(List<FeaturePoint>
newFacePoints, Person person)
     {
       var personPhotos = photos.FindAll(x => x.idPerson == person.id);
```

```
      var personSimilarity = 1 / (1 + measureDistance(newFacePoints,
centroid(person)));

      return new DetectedPerson
      {
        person = person,
        percentOfSimilarity = personSimilarity,
        photo = photos.First()
      };
    }
```

Метод «measureSimilarityCentroid» за допомогою бібліотеки ланцюжкової обробки колекцій виконує пошук усієї відеоінформації певної особи. За допомогою описаного вище методу «measureDistance» обчислюється відсоток подібності ХТ нової відеоінформації до усередненого за формулою центроїда (3.23) списку ХТ.

Реалізацію формули (3.23) наведено нижче:

```
List<FeaturePoint> centroid(Person person)
  {
   var personPhotos = photos.FindAll(x => x.idPerson == person.id);
   var summed = centroid(personPhotos[0]);
   for (var i=1; i<personPhotos.Count; ++i)
   {
    var next = centroid(personPhotos[i]);
    for (var j=0; j<summed.Count; j++)
    {
     summed[j].x += next[j].x;
     summed[j].y += next[j].y;
    }
   }
   for (var j = 0; j < summed.Count; j++) {
    summed[j].x /= personPhotos.Count;
```

```
    summed[j].y /= personPhotos.Count;
   }
   return summed;
  }

  List<FeaturePoint> centroid(Photo photo)
  {
   return featurePoints.FindAll(x => x.idPhoto == photo.id)
    .OrderBy(x => x.pointPosition).ToList();
  }
```

Формула (3.24) для класифікації об'єктів методом суми описана наступним фрагментом програми:

```
private DetectedPerson measureSimilaritySum(List<FeaturePoint>
newFacePoints, Person person)
  {
   var personPhotos = photos.FindAll(x => x.idPerson == person.id);
   var personSimilarity = 1 / ( 1 +
    personPhotos.Select(photo => measureDistance(newFacePoints,
photo)).Sum());

   return new DetectedPerson {
    person = person,
    percentOfSimilarity = personSimilarity,
    photo = photos.First()
   };
  }
```

Цікавою особливістю реалізації цієї функції є використання ланцюжка з двох перетворень вхідної множити відеоінформації, а саме перетворення ХТ відеоінформації у значення відстані оператором Select за допомогою описаної вище функції «measureDistance» та обчислення суми цих відстаней за допомогою оператора Sum.

Формула (3.25) для класифікації об’єктів методом максимуму описана наступним фрагментом програми:

```
private DetectedPerson measureSimilarityMax(List<FeaturePoint>
newFacePoints, Person person)
  {
   var personPhotos = photos.FindAll(x => x.idPerson == person.id);
   var personWithSimilarity = personPhotos.Select(photo => new
DetectedPerson
   {
    person = person,
    photo = photo,
    percentOfSimilarity = 1 / (1 + measureDistance(newFacePoints, photo))
   }).OrderBy(x => x.percentOfSimilarity).LastOrDefault();

   return personWithSimilarity ?? new DetectedPerson { person = person,
percentOfSimilarity = 0 };
  }
```

У методі «measureSimilarityMax» виконується перетворення множини відеоінформації з БД у об’єкти «DetectedPerson» в такий спосіб, що на виході отримуємо значення подібності доповнене назвою файлу відеоінформації, де виявлено шуканий об’єкт з найвищою точністю.

Метод «toFeaturePoints» реалізує нормування вектора ХТ за формулами (3.13)-(3.16):

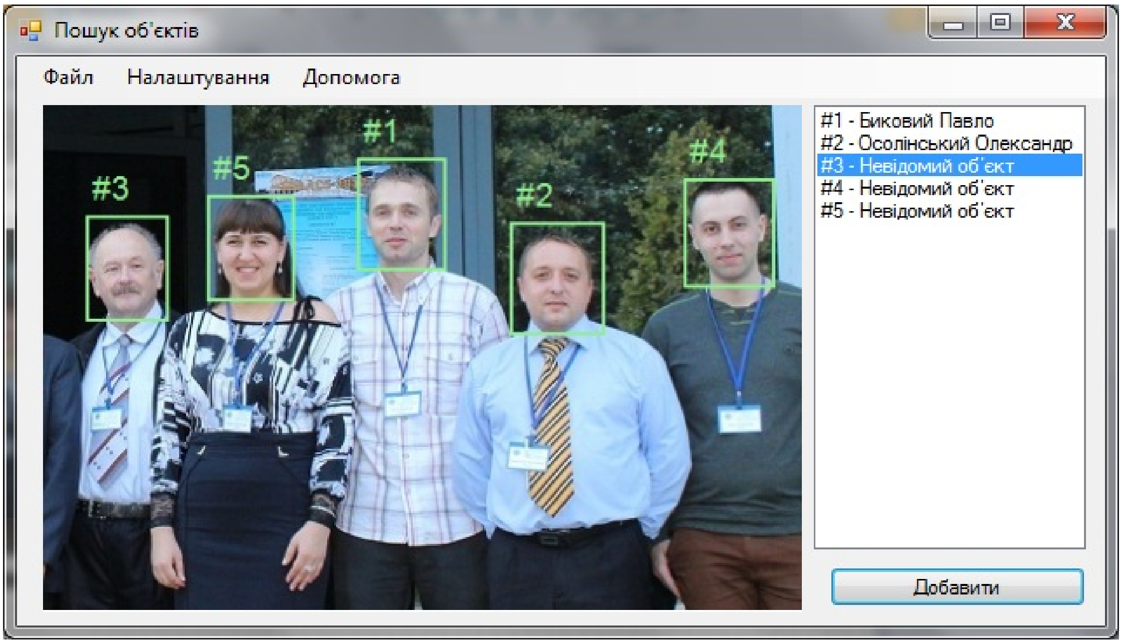
```
public static List<FeaturePoint> toFeaturePoints(List<Point> points, int
idPhoto)
  {
   var featurePoints = new List<FeaturePoint>();
   var sumPoint = new Point(0, 0);
```

```
  foreach (var point in points)
  {
   sumPoint.X += point.X;
   sumPoint.Y += point.Y;
  }
  var centerX = (double)sumPoint.X / points.Count;
  var centerY = (double)sumPoint.Y / points.Count;

  var scale = 0.5 * Math.Abs(points[16].X - points[0].X);

  for(var i=0; i<points.Count; i++)
  {
   featurePoints.Add(new FeaturePoint
   {
    idPhoto = idPhoto,
    pointPosition = i,
    x = (points[i].X - centerX) / scale,
    y = (points[i].Y - centerY) / scale
   });
  }

  return featurePoints.OrderBy(x => x.pointPosition).ToList();
 }
```

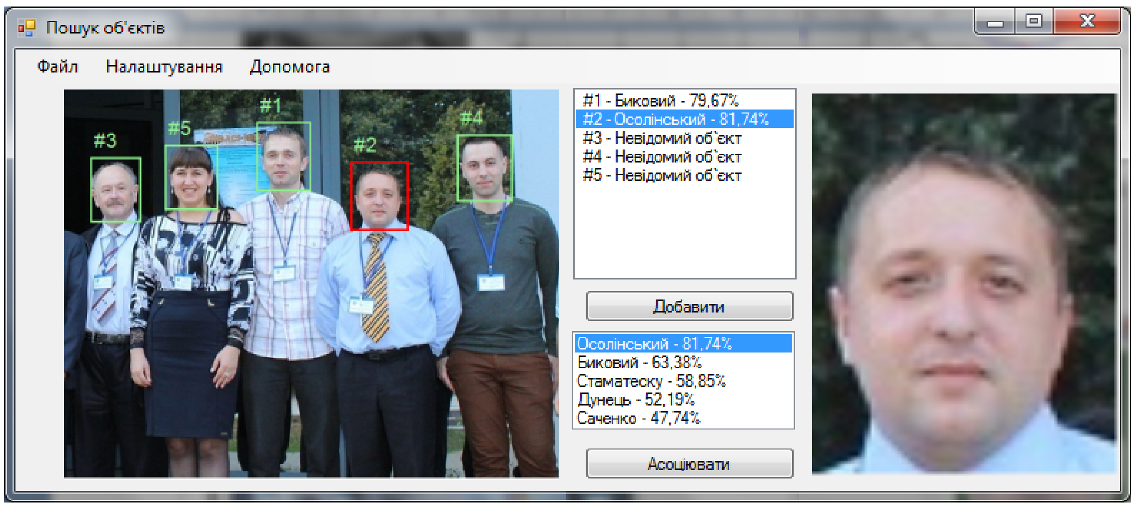
Розглянемо користувацький інтерфейс розробленої ІТ АСВН. Вона складається з вікна вводу відеоінформації, вікна вводу нових об’єктів та вікна перегляду та редагування бази існуючих об’єктів.

Вікно пошуку об’єктів (рисунок 4.8) базового та розширеного вигляду дозволяє відкрити файл відеоінформації, на котрому модуль виявлення облич [118] виділить прямокутником обличчя об’єктів та їх порядковий номер.

До виділених об’єктів застосовується метод КС на базі методу Канні для пошуку контурів для отримання послідовності контурних препаратів; для кожного контуру проводиться виділення ХТ за допомогою методу вейвлет-аналізу функції кривизни, будується ідентифікаційний вектор на основі якого проводиться класифікація на кожному рівні деталізації, і на наступному рівні класифікації проводиться порівняння тільки з тими об'єктами, які пройшли класифікацію на попередньому рівні. На останньому рівні виводиться прізвище та ім'я об'єкта, з яким найбільший відсоток співпадінь. Якщо рівень співпадіння низький (менший за встановлений в налаштуваннях поріг), об'єкт позначається як «# номер - невідомий об'єкт».



а)



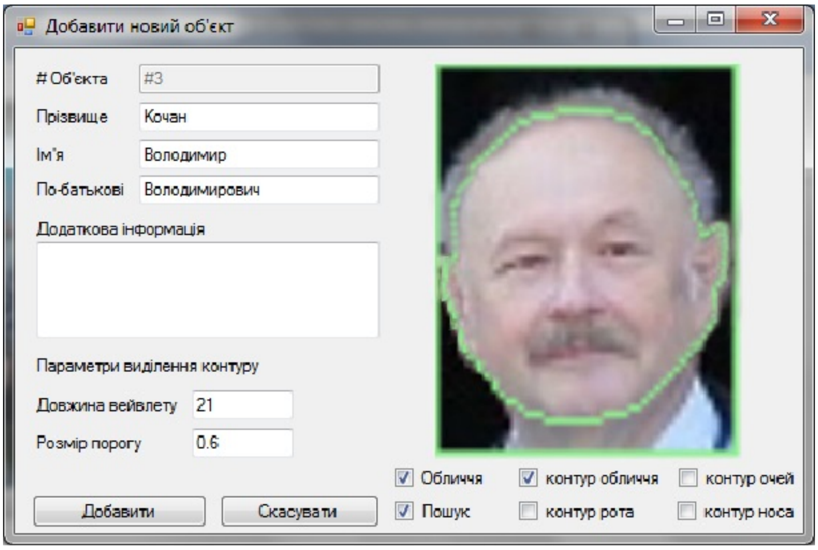
б)

Рисунок 4.8 – Вікна ідентифікації об'єктів АСВН:

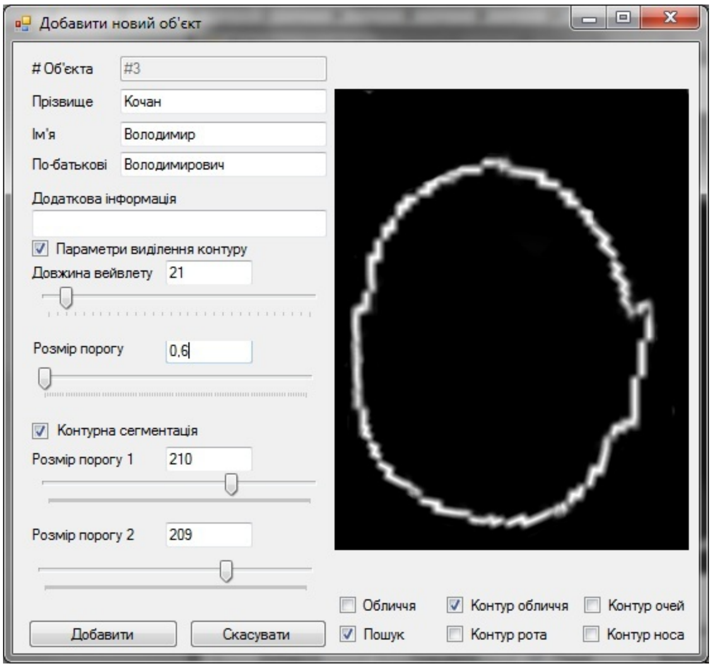
а) базовий вигляд, б) розширений вигляд з інформацією про виявлені обличчя

Вікно розширеного вигляду дозволяє додатково переглянути відсоток подібності наявних на відеоінформації облич до присутніх у БД об'єктів, впорядкованих за спаданням відсотка подібності. Вибір об'єкта дозволяє переглянути його зображення у формі праворуч (Рисунок 4.8 (б)).

В залежності від встановлених налаштувань системи, після вибору невідомого об'єкта та натиснення кнопки «Добавити» можна відкрити спрощену форми вводу нових об'єктів (рисунок 4.9).



а)



б)

Рисунок 4.9 – Вікна вводу нового об'єкта в БД АСВН:

а) базовий вигляд, б) розширений вигляд

У них можна вказати прізвище, ім’я, по-батькові, встановити довжину вейвлета і розмір порогу, пороги контурної сегментації, переглянути виділені контури та добавити відповідний об’єкт в систему.

Вибір кнопки «Асоціювати» форми з рисунку 4.4 (б) дозволяє віднести невиявлений об’єкт до існуючого кластеру в БД.

Вибір в меню опції «Перегляд об’єктів» викликає форму перегляду та редагування кластерів відеоінформації (рисунок 4.6). Кожен кластер відеоінформації описується прізвищем особи, що йому відповідає. Вибір прізвища призводить до завантаження списку відеоінформації, що асоційована із ним.

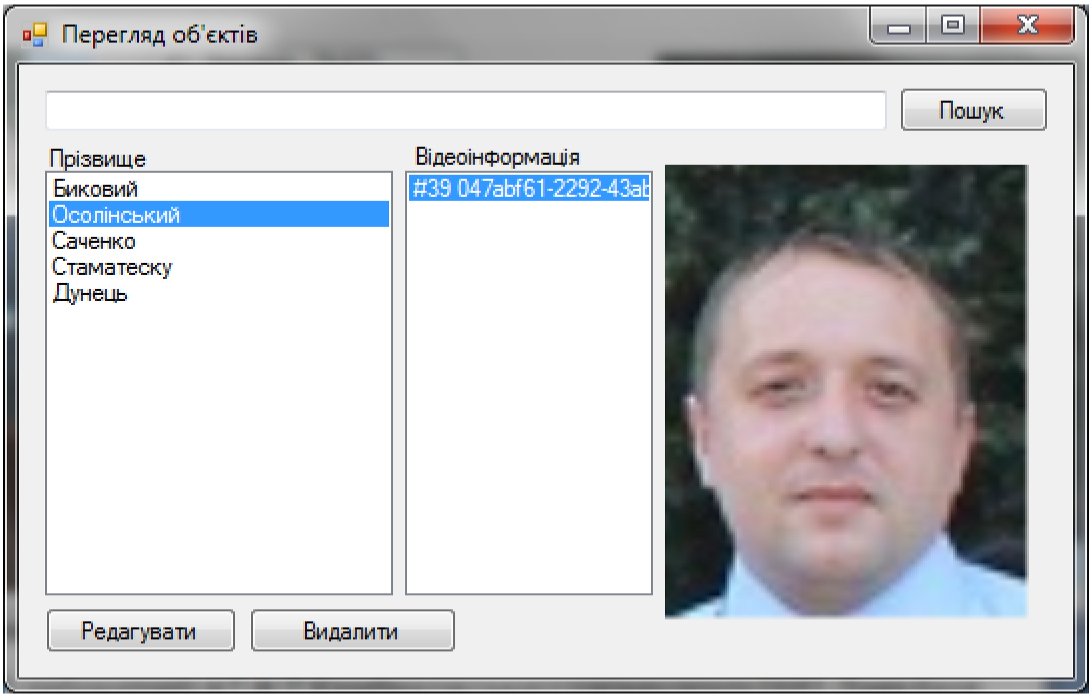


Рисунок 4.10 – Форма редагування та пошуку об’єктів в БД

4.3. Експериментальні дослідження інформаційної технології

Для перевірки роботи АСВН використано зображення облич з бази даних зображень ORL лабораторії AT & T Кембриджського університету [89] та базу облич CMU PIE [119].

База облич ORL містить набір зображень облич зроблених з квітня 1992 року до квітня 1994 року в лабораторії. Вона використовувалась для проекту з розпізнавання облич, що виконувався спільно з Vision and Robotics Group в інженерному департаменті Кембриджського університету. База зображень ORL складається із десяти різних зображень сукупності з 40 різних осіб. Для деяких осіб, зображення були зроблені в різний час, із зміненим освітленням, зміненою мімікою (відкритими або закритими очима, у посмішці або без посмішки), зміненими рисами обличчя (окуляри чи без окулярів). Всі знімки бази зображень були зроблені на темному однорідному фоні з обличчями у вертикальній фронтальній позиції (деякі із поворотом чи нахилом у якийсь бік). Формат файлів – PGM, розмір кожного зображення 76x92 пікселів, з 256 рівнями відтінків сірого. Зображення організовані в 40 каталогах (по одному на кожну особу), які мають імена виду SX , де x вказує на номер (від 1 до 40).

База даних облич CMU PIE що містить понад 40 тис. кольорових зображень 68 осіб (640x480, колір 24-біт).

В таблиці 4.1 виконано дослідження запропонованих в § 3.3 методів класифікації об’єктів (на основі центроїда, суми і максимуму) та схем для обчислення відстані між векторами ХТ (класичною евклідовою метрикою та модифікованою на основі відстаней від початку координат) на вибірці 100 людей, з яких 10 вважали підозрілими.

Для оцінювання значень помилок класифікації вибрано два об’єкти: перший являє собою відеоінформацію, де зображено обличчя у розшуку, а другий – відеоінформацію, де зображено обличчя, що не знаходяться у розшуку (звичайна особа).

Помилки класифікації поділяються на помилки першого (I) та другого (II) роду [120]. Помилка першого роду – це невиявлення розшукуваного обличчя (об’єкти 1) на відеоінформації, де апріорі відомо, що воно там є. Помилка другого роду –хибна класифікація обличчя як такого, що знаходиться у розшуку, коли апріорі відомо, що це не так (об’єкти 2).

З таблиці видно, що із збільшенням кількості осіб у кадрі погіршується достовірність по помилках II роду (звичайних людей віднесено до підозрілих).

Якщо збільшувати кількість людей в кадрі у 8-10 разів, то запропонована АСВН швидко приймає рішення, проте вони не завжди достовірні.

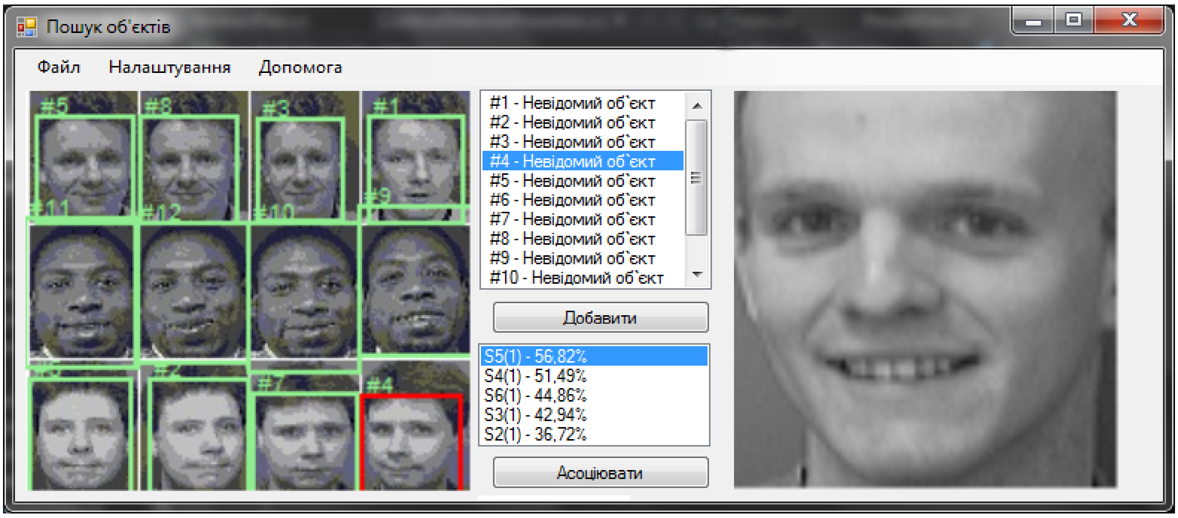
Такий результат можна пояснити низькою роздільною здатністю відеоінформації, що вміщує таку велику кількість облич, в такому випадку оператор наводить телевізійний датчик ще раз та приймає остаточне рішення.

Таблиця 4.1 – Порівняння співвідношень методів класифікації.

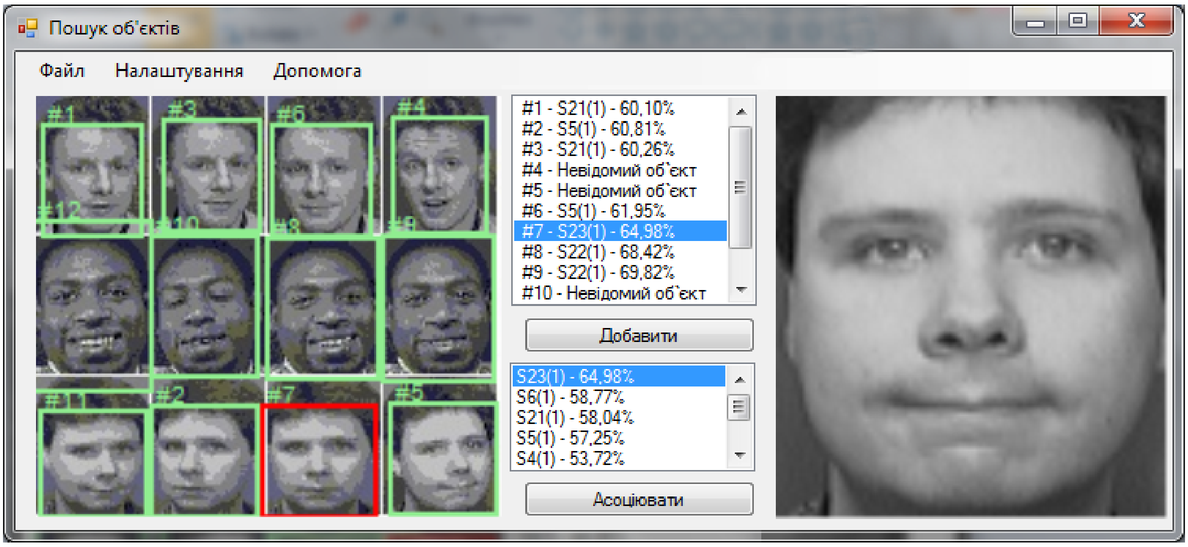
Виявлення обєкта 1 на кадрі з 4 осіб				Виявлення обєкта 2 на кадрі з 4 осіб			
	Метод класифікації				Метод класифікації		
Метод оцінювання відстані між векторами ХТ	центроїд (3.22)	сума (3.24)	максимум (3.25)	Метод оцінювання відстані між векторами ХТ	центроїд (3.22)	сума (3.24)	максимум (3.25)
класичний (3.19)	66.3%	56.1%	73.0%	класичний (3.19)	12.5%	20.6%	10.1%
модифікований (3.20)	65.5%	57.2%	78.4%	модифіковани й (3.20)	9.3%	15.6%	3.4%
Виявлення обєкта 1 на кадрі з 20 осіб				Виявлення обєкта 2 на кадрі з 20 осіб			
	Метод класифікації				Метод класифікації		
Метод оцінювання відстані між векторами ХТ	центроїд (3.22)	сума (3.24)	максимум (3.25)	Метод оцінювання відстані між векторами ХТ	центроїд (3.22)	сума (3.24)	максимум (3.25)
класичний (3.19)	58.2%	43.8%	69.8%	класичний (3.19)	17.3%	31.2%	22.4%
модифікований (3.20)	61.0%	49.0%	65.3%	модифіковани й (3.20)	14.5%	16.5%	5.3%
Виявлення обєкта 1 на кадрі з 40 осіб				Виявлення обєкта 2 на кадрі з 40 осіб			
	Метод класифікації				Метод класифікації		
Метод оцінювання відстані між векторами ХТ	центроїд (3.22)	сума (3.24)	максимум (3.25)	Метод оцінювання відстані між векторами ХТ	центроїд (3.22)	сума (3.24)	максимум (3.25)
класичний (3.19)	43.3%	35.6%	51.1%	класичний (3.19)	23.3%	33.4%	30.1%
модифікований (3.20)	65.5%	57.2%	58.2%	модифіковани й (3.20)	19.4%	22.3%	17.1%

Також з таблиці можна зробити висновок, що найякісніші результати класифікації дає співвідношення методів оцінювання відстані між векторами ХТ за модифікованою евклідовою метрикою (3.20) та методу класифікації максимумом (3.25).

Приклад роботи АСВН із базою облич ORL наведений на рисунку 4.11, де Було завантажено для ідентифікації по чотири різних зображення трьох різних зображень облич яких ще не було внесено в базу(рисунок 4.9 (а)) Оскільки шукані обличчя на внесені в БД, система відображає найбільш подібне з існуючих. Після завантаження по одному з чотирьох облич та вибору наступного зображення, де присутні подібні обличчя даних осіб, система вивела (рисунок 4.9 (б)).



а)



б)

Рисунок 4.11 – Вікна ідентифікації об'єктів АСВН

З метою оцінювання часової складності методів виділення ХТ та класифікації облич, виконано дослідження того, як застосування процедур структурно-статистичної ідентифікації ієрархічних об’єктів позначується на тривалості класифікації облич у кадрах відеоінформації. В таблиці 4.2. наведено результати цього дослідження.

Таблиця 4.2 – Порівняння тривалостей обробки кадру з різною кількістю об’єктів.

Кількість об’єктів у кадрі	Тривалість обробки кадру з використанням (Так) та без використання (Ні) ієрархічного методу, сек.		Скорочення тривалості за рахунок використання методу, %
	Ні	Так	
1	0,05	0,75	1500
3	0,08	0,96	1200
5	0,1	1	1000
10	0,3	2,4	800

В таблиці 4.3. показано достовірність відповідних результатів класифікації з використанням (Так) та без використання (Ні) процедур структурно-статистичної ідентифікації ієрархічних об’єктів.

Таблиця 4.3 – Порівняння достовірності класифікації об'єктів кадру з різною кількістю об’єктів.

Кількість об’єктів у кадрі	Достовірність класифікації з використанням (Так) та без використання (Ні) ієрархічного методу, %		Різниця достовірності, %
	Ні	Так	
1	100	95	5
3	95	88	7
5	93	85	8
10	90	80	10

Як видно з таблиць, застосування процедур структурно-статистичної ідентифікації ієрархічних об’єктів в інформаційній технології відеонагляду дозволило зменшити час обробки кадру у 8-15 разів (в залежності від вмісту кадру та кількості об'єктів на ньому) при невеликому зниженні достовірності (5-10%).

Таким чином, у даному підрозділі впроваджено ІТ контурної сегментації, виділення ХТ контурів та ідентифікації у АСВН. Впровадження цих ІТ дозволило зменшити час обробки відеокадру.

Висновки до розділу 4

У четвертому розділі на базі створених методів та базових ІТ контурної сегментації, виділення ХТ контуру з регульованою деталізацією та структурно-статистичної ідентифікації розроблено прикладні ІТ для системи підтримки прийняття рішень про форму об'єктів ієрархічної структури в АСВН, що дозволило підвищити їх оперативність.

Запропоновані методи було реалізовано та досліджено в середовищі Matlab з використанням бібліотек Image Processing Toolbox та Curve Fitting Toolbox, зокрема методи виділення ХТ контуру, контурної сегментації на базі методу Канні та структурно-статистичної ієрархічної ідентифікації.

Ідентифікація об'єктів за обличчями проводиться на декількох рівнях: на першому рівні об'єкт відноситься до певного класу за формою голови. Якщо даної інформації не достатньо, то переходимо на наступний рівень ієрархії об'єкту (ідентифікаційні вектори контурів носа, очей та рота).

На базі запропонованих у розділах 2 і 3 методів, в рамках науково-дослідної роботи № 0112U008453 було розроблено АСВН, яку програмно реалізовано в середовищі Visual Studio 2015 на мові програмування C# з використанням бібліотек OpenCV.

Дані ІТ впроваджено у АСВН у ПП «ЕКОДЕРІ» , ТОВ “ВІП комп'ютер”, ТзОВ завод «РЕМПОБУТТЕХНІКА», а також використано у навчальному процесі (додаток И).

Розроблена АСВН у режимі тестування показала зменшення часу обробки кадру у 8-15 разів, в залежності від кількості облич в кадрі. Проте на вибірці 100 людей, з яких 10 вважали підозрілими знизилась достовірність розпізнавання за помилками II роду на 5-10%.

ВИСНОВКИ

У дисертаційній роботі вирішено актуальну науково-прикладну задачу підвищення швидкодії автоматизованих систем відеонагляду в умовах обмежених обчислювальних ресурсів. Для вирішення цієї задачі розроблено методи та базові інформаційні технології контурної сегментації, виділення характерних точок контуру та структурно-статистичної ідентифікації для ієрархічної ідентифікації та класифікації об'єктів. Це дало можливість скоротити час роботи автоматизованих систем відеонагляду, і як наслідок, підвищити оперативність таких систем.

1. Проаналізовано існуючі системи відеонагляду та встановлено, що швидкість їхньої роботи можна підвищити. Проаналізовано методи виділення контурів та встановлено, що існуючі методи не дозволяють отримати послідовність контурних препаратів ієрархічного об'єкта з деталізацією, яка регулюється. Проаналізовано методи та базові інформаційні технології виділення характерних точок контуру та встановлено, що в існуючих методах не регулюється кількість виділення характерних точок у відповідності з метою обробки.

2. Для підвищення оперативності автоматизованих систем відеонагляду запропоновано використати ієрархічний підхід до процедур сегментації та ідентифікації геометричних об'єктів. Для реалізації ієрархічного підходу були вибрані методи вейвлет-аналізу.

3. Розроблено метод та базову інформаційну технологію контурної сегментації на базі методу Канні. Експериментальні дослідження показали його високу завадостійкість за критерієм Према (у 1,2 – 1,5 рази вищу у порівнянні з методом Канні). Розроблений метод та базова ІТ дозволяє представити контурний препарат ієрархічного об'єкта послідовністю контурних препаратів у вигляді об'єкт-підоб'єкт.

4. Розроблено і досліджено метод та базову інформаційну технологію виділення характерних точок контуру з регульованою деталізацією. Експериментальні дослідження показали, що розроблений метод та базова

інформаційна технологія суттєво перевершує диференціальний метод по завадостійкості у 1,2 – 2 рази, а інтерполяційний метод по точності у 1,5 – 3 разів.

5. Розроблено та досліджено метод та базову інформаційну технологію структурно-статистичної ідентифікації об’єктів на базі застосування ієрархічної послідовності ідентифікаційних векторів. Даний метод забезпечує інваріантність до геометричних перетворень об’єктів (зсуву в полі зору, повороту об’єкта та масштабування), володіє високою завадостійкістю, а також високою обчислювальною ефективністю в умовах обмежених обчислювальних ресурсів.

6. Розроблено інформаційну технологію структурно-статистичної ідентифікації ієрархічних об’єктів за характерними точками контурів. Розроблена інформаційна технологія використовується в комп’ютерній системі відеонагляду. Дослідження показали, що застосування цих процедур в інформаційній технології відеонагляду дозволило зменшити час обробки кадру у 8-15 разів (в залежності від вмісту кадру та кількості об’єктів на ньому) при невеликому зниженні достовірності (5-10%).

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Крылов В.Н. Распознавание визуальной информации в автоматизированных системах / В.Н. Крылов, В.Г. Абакумов, С.Г. Антощук // Электроника и связь. – Киев, 2003. – № 19 – С. 46 – 48.

2. Антощук С.Г. Определение геометрических идентификационных характеристик в системе компьютерного распознавания лиц / С.Г. Антощук, О.Ю. Бабилунга // Искусственный интеллект – ПІШІ МОН і НАН України “Наука і освіта”. – 2006. – № 4. – С. 664-669.

3. Антощук С.Г. Модели представления событий при анализе видеопотока / С.Г. Антощук, Н.А. Годовиченко// Електротехнічні та комп’ютерні системи – Одеса, 2013. – № 11. – С. 142-149.

4. Антощук С.Г. Анализ видеопоследовательности при охране экспозиций / С.Г. Антощук, Н.В. Коваленко // Праці Одеського політехнічного університету – Одеса, 2013. – № 1. – С. 43-48.

5. Антощук С.Г. Отслеживание объектов интереса при построении автоматизированных систем видеонаблюдения за людьми / С.Г. Антощук, М.В. Коваленко, М.А. Годовиченко // Электротехнические и компьютерные системы – Одеса, 2012. – № 8(84). – С. 151-156.

6. Щербакова Г.Ю. Исследование автоматизированной кластеризации с использованием вейвлет-преобразования / Г.Ю. Щербакова, В.Н. Крылов, Р.А. Писаренко, О.В. Логвинов // Системи обробки інформації – Харків, 2016. – № 2. – С. 73-75.

7. Крылов В.Н. Вторичные преобразователи сигналов зображений / Крылов В.Н., Максимов М.В. – Одесса: Астропринт, 1997р. – 176с.

8. Полякова М.В. Морфологический метод контурной сегментации изображений на основе репагулярного вейвлет-преобразования / Полякова М.В., Крылов В.Н. // Труды Одесского политехнического университета – 2006. – Вып. 1(25). – С. 98 – 103.

9. Polyakova M. Image contour segmentation in the space of coefficients of signal semantic wavelet transform / M. Polyakova, V. Krylov // Modern Problems of Radioengineering, Telecommunications and Computer Science: Proc. of Intern. Conf. TCSET 2008, Ukraine – 2008. – P. 280–283.

10. Polyakova M. Classification of methods of the signal semantic wavelet transform for image contour segmentation / M. Polyakova, V. Krylov // Int. Journal of Computing – 2008. – Vol. 7 – Issue 1– P. 51 – 57.

11. Путятін Є.П. Методи та алгоритми комп’ютерного зору: навчальний посібник / Є.П. Путятін, В.О. Гороховатський, О.О. Матат. – Х.: ТОВ «Компанія СМІТ», 2006. – 236 с.

12. Гороховатский В.А. Структурное распознавание изображений на основе моделей голосования признаков характерных точек / В.А. Гороховатский, Е.П. Путятин // Реєстрація, зберігання і обробка даних. – 2008. – Т.10. - № 4. – С. 75-85.

13. Mashtalir V. Clustering Method for Fast Content-Based Image Retrieval / V. Mashtalir, D. Kinoshenko, E. Yegorova // Computational Imaging and Vision / M.A. Viergever, ed. – Dordrecht: Springer. – 2006. – Vol. 32. – P. 946–952.

14. Березький О.М. Метод вибору алгоритмів автоматичної сегментації біомедичних зображень / О.М. Березький, Ю.М. Батько // Системи обробки інформації. – 2013. - № 2 – С. 246-251.

15. Березький О. Методи кількісної оцінки якості сегментації зображень / О. Березький // Матеріали дванадцятої всеукраїнської міжнародної конференції «Оброблення сигналів і зображень та розпізнавання образів» (УкрОБРАЗ’2014), Київ, 3-7 листопада 2014 р. – К., 2014. –С. 51-54.

16. Грицик В. Методи та високопродуктивні паралельні системи для обробки та розпізнавання зображень у реальному часі / В.В. Грицик, О.М. Березький // International Journal of Computing. – 2014. – Том 2 – Вип. 1 (№ 3) – С. 25-34.

17. Русин Б.П. Системи розпізнавання образів з малими базами даних / Б.П. Русин, Б.О. Капустій, В.А. Таянов. – Львів: СПОЛОМ, 2006. – 152 с.

18. Керод Т. І. Порівняння завадостійкості методів сегментації зображень за імпульсного зашумлення та спотворення розмиттям / Т.І. Керод, Р.Я. Косаревич, Б. П. Русин, О. В. Капшій, В. В. Корній // Відбір і обробка інформації. - 2015. - Вип. 42. - С. 68-74.

19. Клювак А. Двоетапне сегметування зображення із складним фоном на основі методу Отсу / А. Клювак, Д. Пелешко, О. Маковейчук, І. Ізонін // Вісник Національного університету Львівська політехніка. Комп'ютерні науки та інформаційні технології – 2016. – Вип. 843 – С. 335-341.

20. Пелешко Д. Інваріантні моменти в прикладних задачах обробки та аналізу зображень / Д. Пелешко, А. Ковальчук, Н. Кустра, І. Ізонін // Вісник Національного університету "Львівська політехніка". – 2011. – № 694 : Комп'ютерні науки та інформаційні технології. – С. 265-270.

21. Шлезингер М.И. Десять лекций по статистичному и структурному распознаванию образов / М.И. Шлезингер, В. Главач. – К.: Наук. думка, 2004. – 545 с.

22. Білинський Й.Й. Методи обробки зображень в комп'ютеризованих оптико-електронних системах: монографія / Й.Й. Білинський – Вінниця: ВНТУ.– 2010.–272 с.

23. Білинський Й.Й. Метод виділення контуру на слабоконтрасних розмитих зображеннях / Й.Й. Білинський, І.В. Микулка // Вісник Вінницького політехнічного інституту. – 2012. - № 3. – С. 164-169.

24. Schimid C. Evaluation of Interest Point Detectors [Text] / C. Schimid, R. Mohr, C. Bauckhane // International Journal of Computer Vision. – 2nd ed. – 2000. – Vol. 37. – P. 151-172.

25. Tarabalka Yu. Improved hierarchical optimization-based classification of hyperspectral images using shape analysis / Yuliya Tarabalka, James C. Tilton // IEEE International Geoscience and Remote Sensing Symposium – 2012 - P. 1409-1412.

26. Прэт У. Цифровая обработка изображений: в 2-х книгах / Прэт У.; пер. с англ. Д.С. Лебедев. – М.: Мир, 1982. – Кн. 2 – 480 с., ил.

27. Jun Liang. Salient feature point detection for image matching / Jun Liang, Yanning Zhang, Steve Maybank, Xiuwei Zhang // IEEE China Summit & International Conference on Signal and Information Processing (ChinaSIP). – 2014 – P. 485 – 489.

28. Daguang Jiang Comparion and Study of Classic Feature Point Detection Algorithm / Daguang Jiang, Junkai Yiγ // Proceedings of the International Conference on Computer Science and Service System. – 2012 – P. 2307-2309.

29. Gonzalez R. C. Digital Image Processing, 3rd Edition, // R. C. Gonzalez, R.E. Woods - Pearson, 2007. – 976 p.

30. Фурман Я. А. Введение в контурный анализ: приложения к обработке изображений и сигналов [Текст] / Я. А. Фурман, А. В. Кревецкий, А. А. Роженцов, Р. Г. Хафизов, И. Л. Егошина, А. Н. Леухин ; под ред. Я. А. Фурмана. – 2-е изд., испр. – М.: ФИЗМАТЛИТ, 2003. – 592 с.

31. Загородняя Д.И. Метод идентификации лиц по характерным точкам контура / Д.И. Загородняя // Вестник Брестского государственного технического университета. Серия физика, математика, информатика. – 2015. – №5. – С. 30-33.

32. Загородня Д.І. Аналіз інтерполяційного методу виділення характерних точок контуру / Д.І. Загородня // Науковий вісник Чернівецького національного університету ім. Ю. Федьковича. Серія: Комп’ютерні системи та компоненти. – 2015. – Том 6, Випуск 2. – С. 88-92.

33. Загородня Д.І. Порівняння методів виділення характерних точок контуру / Д.І. Загородня // Радіоелектронні і комп’ютерні системи. Харків «ХАІ». – 2016. – № 3 (77). – С. 56-61.

34. Загородня Д.І. Налаштування параметрів алгоритму виділення характерних точок на основі вейвлету Хаара / Д.І. Загородня // Сучасні комп’ютерні інформаційні технології: матеріали V Всеукраїнської школи-семінару молодих вчених і студентів ACIT'2015. – Тернопіль: ТНЕУ, 22-23 травня 2015. – С. 68-70.

35. Загородня Д.І. Підхід до структурно-ієрархічної контурної сегментації / Д.І. Загородня // Інформатика, управління та штучний інтелект. Матеріали другої

науково-технічної конференції студентів, магістрів та аспірантів. – Харків: НТУ «ХПІ», 26-27 листопада 2015. – С. 31.

36. Загородня Д.І. Дослідження диференціального методу виділення характерних точок контуру облич / Д.І. Загородня. В.І. Дорош, Н.С. Добровольська, О.Л. Римар // Міжнародний науково-технічний журнал «Вимірювальна та обчислювальна техніка в технологічних процесах». – Хмельницький, 2014. – №4 (49). – С. 162-166.

37. Zahorodnia D.I. Canny-based Method of Image Contour Segmentation / Diana Zahorodnia, Yuriy Pigovsky, Pavlo Bykovyy // International Journal of Computing. Ternopil: TNEU. – 2016. – Vol. 15. – Issue 3. – P. 200-205.

38. Загородня Д.І. Метод виділення контурів зображень з використанням вейвлет-перетворень / Д.І. Загородня, У. Гросман У., В.М. Крилов, В.І. Дорош // Науковий вісник Чернівецького національного університету ім. Ю. Федьковича. Серія: Комп’ютерні системи та компоненти. –2016. – Том 7, Випуск 2. – С. 136-141.

39. Paliy Ihor. Fast and Robust Face Detection and Tracking Framework / Ihor Paliy, Volodymyr Dovgan, Ognian Boumbarov, Stanislav Panev, Anatoly Sachenko, Yuriy Kurylyak, Diana Zagorodnya // Proceedings of the IEEE 6th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS’2011). – Prague (Czech Republic), September 15-17, 2011. – Vol.1. – P. 430-433.

40. Zahorodnia Diana. Structural Statistic Method Identifying Facial Images by Countour Characteristic Points / Diana Zahorodnia, Yuriy Pigovsky, Pavlo Bykovyy, Viktor Krylov, Ihor Paliy, Igor Dobrotvor // Proceedings of the 8th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS’2015). – Warsaw (Poland), September 24-26, 2015. – P. 293-297.

41. Базелюк Д.В. Підхід до розпізнавання емоцій на основі зображення обличчя. / Д.В. Базелюк, Д.І. Загородня, І.О. Палій // Труди XIII міжнародної

науково-практичної конференції «Сучасні інформаційні та електронні технології» («СІЕТ-2012»). – Одеса, 4-8 червня 2012. – С. 118.

42. Загородня Д.І. Алгоритми прослідковування контурів та виділення характерних точок в процедурі сегментації системи комп'ютерного розпізнавання / Д.І. Загородня, І.О. Палій, В.М. Крилов // Сучасні комп'ютерні інформаційні технології: Матеріали ІІІ Всеукраїнської школи-семінару молодих вчених і студентів (ACIT'2013). – Тернопіль: ТНЕУ, 17-18 травня 2013. – С.129-130.

43. Загородня Д.І. Метод розпізнавання облич за характерними точками контуру в комп'ютерних системах відеоспостереження. / Д.І. Загородня, І.О. Палій, В.М. Крилов // Труди XIV міжнародної науково-практичної конференції «Сучасні інформаційні та електронні технології» («СІЕТ-2013»). – Одеса, 27-31 травня 2013. – С. 49-52.

44. Загородня Д.І. Алгоритм контурної сегментації в системах відеоспостереження / Д.І. Загородня, К.Ю. Ковалок, А.О. Саченко, В.М. Крилов // Всеукраїнська наукова Інтернет-конференція «Економічна кібернетика – інженерія економіки (випуск 24)» / Збірник тез доповідей: випуск 24. – Тернопіль: Тайп, 29-30 листопада 2013. – С. 3-5.

45. Zahorodnia Diana. Contour Segmentation Method in Video Surveillance Systems / Diana Zahorodnia, Kostantin Kovalok, Anatoly Sachenko, Viktor Krylov, Sergiy Nychyporuk // Proceedings of the International Conference "Modern Problems of Radio Engineering, Telecommunications and Computer Science" (TCSET'2014). – Lviv-Slavske (Ukraine), February 25 – March 1, 2014. – P. 405.

46. Загородня Д.І. Метод виділення характерних точок зображення на основі вейвлет-аналізу функції кривизни / Д.І. Загородня, К.Ю. Ковалок, В.М. Крилов // Труди XVI міжнародної науково-практичної конференції «Сучасні інформаційні та електронні технології» («СІЕТ-2015») – Одеса, 25-29 травня 2015. – С. 50-51.

47. Zahorodnia D.I. Structural-Hierarchical Principle of Contour Segmentation / D.I. Zahorodnia, P.E. Bykovyy // Current Issues in Modern Technologies: Book of abstracts of the IV International Scientific and Technical Conference of Young

Researchers and Students. – Ternopil: TNTU, 25-26 November 2015. – Vol. 2. – P. 20-21.

48. Загородня Д.І. Контроль доступу до комп’ютера на основі розпізнавання зображення обличчя користувача / Д.І. Загородня, О.І. Губчакевич, В.В. Кисіль. // Актуальні задачі сучасних технологій: збірник тез доповідей V міжнародної науково-технічної конференції молодих учених та студентів,– Тернопіль: ТНТУ, 17–18 листопада 2016. – Том 2. – С. 42-43.

49. Дамьяновски В. CCTV. Библия охранного телевидения / Дамьяновски В. – М.: ООО «ИСС», 2002. – 352с.

50. Wang J., Qimei C., De Z., Houjie B. Embedded Wireless Video Surveillance System For Vehicle // International Conference on Telecommunications, Chengdu, China, 2006.

51. Beymer D., McLauchlan P., Coifman B. , Malik J. A Real-Time Computer Vision System for Measuring Traffic Parameters // IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1997.

52. Micheloni C., Foresti G. L., Snidaro L. A Co-Operative Multicamera System for Videosurveillance of Parking Lots // IEEE Symposium on Intelligent Distributed Surveillance Systems, London, 2003.

53. Barrenetxea G., Ingelrest F., Schaefer G., Vetterli M. Wireless Sensor Networks for Environmental Monitoring: The SensorScope experience // IEEE International Zurich Seminar on Communications, Zurich, 2008.

54. Chen T. H., Wu P. H. and Chiou Y. C. An Early Fire-Detection Method Based on Image Processing // IEEE International Conference on Image Processing, Singapore, 2004.

55. Cutrona L., Vivian W., Leith E., Hall G., "A high-resolution radar combat-surveillance system," IRE Transactions on Military Electronics, Vols. MIL-5, no. 2, pp. 127-131, 2009.

56. Norris C., McCahill M., Wood D. The growth of CCTV: a global perspective on the international diffusion of video surveillance in publicly accessible space // Surveillance and Society, vol. 2, no. 2/3, 2004, pp. 110-135.

57. Fleck S., Strasser W. Smart camera based monitoring system and its application to assisted living //Proceedings of the IEEE, vol. 96, no. 10, 2008, pp. 1698-1714.

58. Brezovan M. A review on vision surveillance techniques in smart home environments / M. Brezovan, C. Badica // Proceedings of the 19th International Conference on Control Systems and Computer Science/ - 2013/ - P. 471 – 478.

59. Patrick R. Surveillance systems for smart homes: A comparative survey / R. Patrick, N. Bourbakis // 21st IEEE International Conference on Tools with Artificial Intelligence. – 2009. – P.248 – 252.

60. Kale P.V. Review of Securing Home Using Video Surveillance / Kale P.V., Sharma S.D. // International Journal of Science and Research (IJSR). – Volume 3. – Issue 5. – May 2014. – P.1150-1154.

61. Krumm J., Harris S., Meyers B., Brumit B., Hale M., Shafer S. Multi-camera multiperson tracking for easy living // IEEE International Workshop on Visual Surveillance, Dublin, 2000.

62. "IP surveillance: the next generation security camera application," July 2005. IP Surveillance Solutions [Online]. Available: ftp://ftp10.dlink.com/pdfs/products/IP_Surveillance_Solutions_Brief.pdf.

63. Foresti G. L., C. S. Regazzoni and R. Visvanathan, "Scanning the issue/technology: Special issue on video communications, processing and understanding for third generation surveillance systems," Proceedings of the IEEE, vol. 89, no. 10, pp. 1355-1367, October 2001.

64. Diehl C. P., "Toward efficient collaborative classification for distributed video surveillance,"Pittsburgh, 2000.

65. Green M. W., "The appropriate and effective use of security technologies in U.S. schools. A guide for schools and law enforcement agencies," 1999.

66. Dramatically Reduce the Cost and Complexity of Video Surveillance Storage. Oracle white paper. February 2015. – 10 p. [Online]. Available: http://www.oracle.com/us/ solutions/oos/tiered-storage-infrastructure/wp-video-surveillance-storage-2288409.pdf.

67. Guan Y., J. Zhang, Y. Shang, M. Wu and Y. Liu, "Special environment embedded surveillance platform," in China-Japan Joint Microwave Conference, Shanghai, 2008.

68. Rahimi M., R. Baer, O. I. Iroezi, J. C. Garcia, J. Warrior, D. Estrin and M. Srivastava, "Cyclops: in situ image sensing and interpretation in wireless sensor networks," in International Conference on Embedded Networked Sensor Systems, New York, 2005.

69. Dias F., P. Chalimbaud, F. Berry, J. Serot and F. Marmoiton, "Embedded early vision systems: implementation proposal and hardware architecture," in Cognitive System for Interactive Sensors, Paris, 2006.

70. Downes I., L. Baghaei Rad and H. Aghajan, "Development of a mote for wireless image sensor networks," in Cognitive systems for Interactive Sensors, Paris, 2006.

71. K. Lu and et al., "Wireless broadband access: WIMAX and beyond - a secure and serviceoriented network control framework for WIMAX networks," IEEE Communication Magazine, no. 45, 2007

72. Cao Z. Y., Z. Z. Ji and M. Z. Hu, "An image sensor node for wireless sensor networks," in International Conference on Information Technology: Coding and Computing, Las Vegas, 2005.

73. Компанія „TiSO Production” (Україна) https://tiso-turnstiles.com/catalog/freeway-turnstiles/sweeper-al.

74. Facial recognition for high security access control verification http://auroracs.co.uk/wp-content/uploads/2015/06/Aurora-FaceSentinel-Datasheet-1506.pdf.

75. Mobile Vision API - Find objects in photos and video, using real-time on-device vision technology. https://developers.google.com/vision/

76. Advanced 2D-to-3D algorithms for face recognition applications. http://www.animetrics.com/

77. Face identification for stand-alone or Web solutions - VeriLook SDK. http://www.neurotechnology.com/verilook.html

78. Acsys FRS SDK. http://www.acsysbiometrics.com/product_sdk.html

79. OpenCV documentation. https://docs.opencv.org/

80. Google Photos - Free storage and automatic organization for all your memories. https://www.google.com/photos/about/

81. Біометрична автоматизована інформаційно-пошукова система для правоохоронних органів. http://www.asia-soft.com/documents/ru_project_obraz.pdf

82. Canny J.E. A computational approach to edge detection / J.E. Canny // IEEE Trans Pattern Analysis and Machine Intelligence. — 1986. — № 8. — P. 679 - 698.

83. Анисимов Б.В. Распознавание и цифровая обработка зображений / Анисимов Б.В., Курганов В.Д., Злобин В.К. – М.: Высшая школа, 1983р. - 295 с.

84. Дуда Р. Распознавание образов и анализ сцен / Р. Дуда , П. Харт; пер. с англ. Г.Г. Вайнштейна и А.М. Васьковского под. ред. В.Л. Стефанюка. – М.: Мир, 1976. – 511с.

85. Полякова М.В. Контурная сегментация зображений с использованием вейвлет-анализа / М.В. Полякова, В.В. Любченко // Труды Одесского политехнического университета. – 2004. - № 1 (21). – С. 1-5.

86. Нгуен К.М. Алгоритмы контурной сегментации и распознавания образов объектов систем технического зрения / Нгуен К.М., Колючкин В.Я. // Научное издание МГТУ им. Н.Э. Баумана «Наука и образование». – 2013. - № 4. – С. 187-200.

87. Фомин Я.А. Статистическая теория распознавания образов / Я.А. Фомин, Г.Р. Тарловский. – М.: РиС, 1986. – 264 с.

88. Clark J.J. Authenticating edges produced by zero-crossing algorithms / J.J. Clark // IEEE Trans. Pattern Analysis Machine Intelligence. – 1989. – Vol. 12. – No. 8. – P. 830-831.

89. База зображень: AT&T Laboratories Cambridge. – http://www.cl.cam.ac.uk/research/dtg/attarchive/facesataglance.html.

90. Ian-Jiun Ding. Compression for the Feature Points with Binary Descriptors / Ian-Jiun Ding, Szu-Wei Fu, Ching-Wen Hsiao, Pin-Xuan Lee, Yen-Chun Chen //

Proceedings of the 19th International Conference on Digital Image Processing. – 2014, Taiwan. – P. 651-656.

91. Daguang Jiang. Comparion and Study of Classic Feature Point Detection Algorithm / Daguang Jiang, Junkai Yiγ // Proceedings of the International Conference on Computer Science and Service System. – 2012. – P. 2307-2309.

92. Neeta Nain. Fast Feature Point Detector / Neeta Nain, Vijay Laxmi, Bhavitavya Bhadviya, Deepak B. M., Mushtaq Ahmed // IEEE International Conference on Signal Image Technology and Internet Based Systems – 2008. – P. 301 - 306.

93. Rodehorst V. Comparison and Evaluation of Feature Point Detectors [Text] / V. Rodehorst, A. Koschan // In Proc. of 5th Int. Symposium Turkish-German Joint Geodetic Days "Geodesy and Geoinformation in the Service of our Daily Life". – Tech. Univ. of Berlin, Germany. – March 2006. – P. 8.

94. Tuytelaars, T. Local Invariant Feature Detectors: A Survey [Text] / T. Tuytelaars, K. Mikolajczyk // Found. and Trends in Computer Graphics and Vision. – 2008. – Vol. 3, No. 3, – P. 177-280.

95. Козина Ю. Ю. Применение алгоритма веерной интерполяции для выделения характерных точек контуров реперных знаков на печатных платах / Ю.Ю. Козина // Збірник наукових праць Військового інституту Київського національного університету імені Тараса Шевченка. - 2014. - Вип. 46. - С. 25-28.

96. Alok Kumar Singh Kushwaha. Performance evaluation of various moving object segmentation techniques for intelligent video surveillance system / Alok Kumar Singh Kushwaha, Rajeev Srivastava // International Conference on Signal Processing and Integrated Networks (SPIN), 2014. - P. 196-201.

97. Демьянович Ю.К. Введение в теорию вэйвлетов. Курс лекций / Демьянович Ю.К., Ходаковский В.А. - Санкт-Петербург, 2007р. – 49с.

98. Krylov V. The Method of Image Contour Segmentation Based on Wavelet Transform and Mathematical Morphology / V. Krylov, M. Polyakova // Modern Problems of Radioengineering, Telecommunications and Computer Science: Proc. of Intern. Conf. TCSET’2006. – Lviv, 2006. – P. 236–239.

99. Квєтний Р.Н. Метод виділення контурів на основі вейвлет-перетворення з використанням двовимірних фільтрів / Р.Н. Квєтний, Є.М. Бочков // Інформаційні технології та комп'ютерна інженерія. – 2010. - № 3. – С. 26-34.

100. Полякова М. Метод построения улучшенных вейвлетов путем преобразования графика степенной функции для задачи контурной сегментации зображений / М. Полякова, В. Крылов, Н. Волкова // Компьютинг. – 2012. - Т.11, Вип. 3. – С. 203-214.

101. Антощук С.Г. Метод морфологической обработки полутоновых изображений на основе гиперболического вейвлет-преобразования / С. Г. Антощук, О. Ю. Бабилунга, А. А. Николенко // Електромашинобуд. та електрообладн. - 2006. - Вип. 67. - С. 119-123.

102. Ногин В.Д. Принятие решений в многокритериальной среде: количественный поход / Ногин В.Д. – М.: ФИЗМАТЛИТ – 2002. – 144 с.

103. Пупков К.А. Функциональные ряды в теории нелинейных систем / Пупков К.А., Капалин В.И., Ющенко А.С.- М.: Наука. – 1976.- 448 с.

104. Тимченко Л.И. Параллельно–иерархические сети как структурно – функциональный базис для построения специализированных моделей образного компьютера: Монография / Л.И. Тимченко, В.П. Кожемяко, А.А. Яровой. – Винница: Универсум - Винница, 2005 – 161 с.

105. Крылов В.Н. Совмещение изображений в системах оптического контроля печатных плат / В.Н. Крылов, Г.Ю. Щербакова // Технология и конструирование в электронной апаратуре. — 2000р.— № 1.— С. 23 - 25.

106. Гайдышев И. Анализ и обработка данных: специальный справочник / И. Гайдышев – СПб: Питер, 2001. 752 с.

107. Гонсалес Р. Принципы распознавания образов / Гонсалес Р., Ту Дж. – М.: Мир. – 1978 г. – 416 с.

108. Карпова Т. С. Базы данных: модели, разработка, реализация / Т. С. Карпова – СПб. : Питер, 2002. – 304 с.

109. Коннолли Т. Базы данных. Проектирование, реализация и сопровождение. Теория и практика / Т. Коннолли, К. Бегг ; Пер. с англ. – 3-е изд. – М. : Издательский дом “Вильямс”, 2003. – 1440 с.

110. Al Najjar M. Video Surveillance for Sensor Platforms: Algorithms and Architectures. Lecture Notes in Electrical Eng. [Text] / M. Al Najjar, M. Ghantous, M. Bayoumi. – Springer, Book 114, 2013. – 202 p.

111. Caputo A. C. Digital Video Surveillance and Security [Text] / A. C. Caputo. – Second edition. – Butterworth-Heinemann, 2014. – 440 p.

112. Kapustiy B.O. Peculiarities of Application of Statistical Detection Criteria for Problems of Pattern Recognition / B.O. Kapustiy, B.P. Rusyn, V.A. Tayanov // Journal of Automation and Information Science. – Vol. 37/ - Issue 2. – 2005. - P 30-36.

113. SQLite - реляційна система керування базами даних https://www.sqlite.org/about.html

114. Пасічник В. В. Організація баз даних та знань / В. В. Пасічник, В.А. Резніченко. – К. : Видавнича група BHV, 2006. – 384 с.

115. Gamma E., Helm R., Johnson R., Vlissides J., Booch G. Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley Professional; 1st ed. 1994, 395 p.

116. McConnell S. Code Complete: A Practical Handbook of Software Construction: A Practical Handbook of Software Costruction Microsoft Press; 2 ed edition, 2004, 960 p.

117. Larman C. Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development. Prentice Hall; 3 ed. 2004, 736 p.

118. Paliy I. Face detection on grayscale and color images using combined cascade of classifiers / I. Paliy // International Journal of Computing. – 2009. – Vol 8. – №. 1. – P. 61-71.

119. Sim T., Baker S., Bsat M. The CMU pose, illumination, and expression (PIE) database. In Proc. of the 5th IEEE International Conference on Automatic Face and Gesture Recognition, 2002. 8 p.

120. Mansfield A. J., Wayman J. L. Best Practices in Testing and Reporting Performance of Biometric Devices // NPL Report CMSC 14/02, August 2002. 32 p.

ДОДАТКИ

Додаток А
Метод контурної сегментації на базі методу Канні та критерій Прета

Лістинг програми методу контурної сегментації

```
function main

close all
name = '9_11_2';
img = imread(sprintf('../Metod_Based_on_Kanny/%s.jpg', name));
if ndims(img) == 3
    img = rgb2gray(img);
end
figure, imshow(img), title('Orig')

n = 8; % довж вейвлета

img = img/max(max(img)); % нормування

img = widening(img, n);%доозначення по горизонталі
img = widening(img', n)'; %доозначення по вертикалі
figure, imshowminmax(img, false), title('widening')

waveletX = [
     -ones(2*n+1,n) zeros(2*n+1,1) ones(2*n+1,n)
 ];
 waveletY = waveletX';

wavImgX = applyWavelet(img, n); % накладання вейвлета по X (в.2)
figure, imshowminmax(wavImgX, false), title('waveletX')
imgThX = thresh(wavImgX, [.5 .5 .3]);
figure, imshowminmax(imgThX, false), title('imgThX')
showWaveletCurves(img, wavImgX)
applyCanny(wavImgX)

wavImgY = applyWavelet(img', n)'; % накладання вейвлета по Y (в.2)
figure, imshowminmax(wavImgY, false), title('waveletY')
imgThY = thresh(wavImgY, [.4 .6 .2]);
figure, imshowminmax(imgThY, false), title('imgThY')
showWaveletCurves(img, wavImgY)
applyCanny(wavImgY)

zzz = double(imgThX > 0 | imgThY > 0);
figure, imshowminmax(zzz, false), title('imgThX or imgThY')

zzz = fillHoles(zzz);
figure, imshowminmax(zzz, false);
title('binary image with filled holes');

BWoutline = bwperim(zzz);
Segout = zeros(size(zzz));
Segout(BWoutline) = 255;
figure, imshow(Segout), title('outlined original image');
```

Лістинг функції накладання вейвлету applyWavelet

```
function img = applyWavelet(orig, n)
img = zeros(size(orig));
for i=1:size(orig,1)
    for j=n+1:size(orig,2)-n
        img(i,j)=abs(-sum(orig(i,j-n:j-1)) + sum(orig(i,j+1:j+n))) / 2/n;
    end
end
```

Лістинг функції реалізації критерія Прета

```
function Pretto

close all
img = imread('P_1.jpg');

sygnNoise = [5 10 20 50 100]

if ndims(img) == 3
    img = rgb2gray(img);
end

figure
imshow(img)
title('Початкове')

a = [5000 1000 500 200 10];
i = 1;

for sn = sygnNoise
    figure
    J=noise(img, a(i) * (1-sn/max(sygnNoise)));
    imshow(J)
    titleStr = sprintf('Зашумлене с/ш = %.f', sn);
    title(titleStr)
    i=i+1;
    canny(J, titleStr)
end

function J=noise(I, n)
J = I;
maxColor = max(max(I));
for i=0:round(n)
    J(randi(size(J,1),1,1), randi(size(J,2),1,1)) = randi(maxColor,1,1);
end
```

Додаток Б
Методи виділення характерних точок та їх дослідження

Лістинг програми інтерполяційного методу

```
close all
I=imread('Kanny-1.JPG');
I = im2bw(I);
figure,imshow(I);

[y,x]=find(I);
y=max(y)-y;

mx=(max(x)-min(x))/2;
my=(max(y)-min(y))/2;

x=x-mx;
y=y-my;

figure, hold off
plot(x,y,'.')

[THETA,RHO] = cart2pol(x,y);
[THETA,ii] =sort(THETA);

x=x(ii);
y=y(ii);

hold on,
inflectionPoints = interpModif(x,y, 4, 16);

plot(x(1),y(1),'ro')
plot(inflectionPoints(:,1), inflectionPoints(:,2),'+-')
```

Лістинг функції interpModif

```
function XT = interpModif(x, y, startN, DELTA)

N = length(x);
x=[x; x];
y=[y; y];

XT = [];

n = startN;
k = 1;

while k < N
    v1 = [x(k) y(k) 0];
    v2 = [x(k+n) y(k+n) 0];
    v = v2-v1;
    u = v/norm(v);

    v_end = v1 + 100*u;
    plot([v1(1) v_end(1)], [v1(2) v_end(2)], 'r:'
    plot([v1(1) v2(1)], [v1(2) v2(2)], 'r-')
```

```
    for i=k : N
        pt = [x(i) y(i) 0];
        delta = point_to_line(pt, v1, v2);
        text(x(i), y(i), num2str(delta))
        if delta > DELTA
            XT=[XT; pt];
            break;
        end
    end
    k = i;
end

XT=[XT; XT(1,:)]

function d = point_to_line(pt, v1, v2)
a = v1 - v2;
b = pt - v2;
d = norm(cross(a,b)) / norm(a);
```

Лістинг програми диференціального методу

```
I=imread('Kann_1.JPG');
I = im2bw(I);

[y,x]=find(I);
y=max(y)-y;

mx=(max(x)-min(x))/2;
my=(max(y)-min(y))/2;

x=x-mx;
y=y-my;

figure(1), hold off
plot(x,y,'.')

[THETA,RHO] = cart2pol(x,y);
[THETA,ii] =sort(THETA);
RHO=RHO(ii);

 [ans, ii] = sort(RHO);

extremums= getExtremums(RHO);
hold on
plot(x(extremums), y(extremums), 'o')


figure(2), hold off

plot(THETA,RHO,'LineWidth',2.5), grid on, hold on
plot(THETA(extremums),RHO(extremums),'ro')
```

Лістинг функції getExtremums

```
function ii = getExtremums(yy)

ii = [];
```

```
going_up = yy(2)>yy(1);
going_down = yy(2)<yy(1);

for i=2:length(yy)
    if yy(i)>yy(i-1) && going_down
        going_down = false;
        going_up = true;
        ii = [ii i-1];
    elseif yy(i)<yy(i-1) && going_up
        going_up = false;
        going_down = true;
        ii = [ii i-1];
    end
end
```

Лістинг програми методу виділення характерних точок на базі вейвлет-аналізу функції кривизни

```
function main

close all, clc

I=imread('Kann_1.JPG');
I = im2bw(I);

[y,x]=find(I);
y=max(y)-y;

mx=(max(x)-min(x))/2;
my=(max(y)-min(y))/2;

x=x-mx;
y=y-my;

figure(1), hold off
plot(x,y,'.')

[THETA,RHO] = cart2pol(x,y);
[THETA,ii] =sort(THETA);

RHO=RHO(ii);
x=x(ii);
y=y(ii);

DELTA = 10;
[ss, ii] = computeSPL(DELTA,THETA,RHO);
figure(2)

plot(THETA(ii), ss, 'b-', 'LineWidth', 2), hold on
plot([THETA(ii(1)) THETA(ii(end))], [0 0], 'k:', 'LineWidth',2)

XT = [];

for i=1:length(ss)-1
    if (ss(i)*ss(i+1)<0) && (isempty(XT) || ii(i)-XT(end)>DELTA)
        [ans, k] = min(abs(ss(i:i+1)))
        if k == 1
            XT=[XT ii(i)];
        else
            XT=[XT ii(i+1)];
```

```
        end
    end
end

figure(1), hold on
plot(x(XT), y(XT), 'ro', 'MarkerSize', 10, 'LineWidth',2)

XT=[XT XT(1)];
plot(x(XT), y(XT), 'k-', 'LineWidth',2)

figure(2), hold on
plot(THETA(XT),0,'ro', 'MarkerSize', 10, 'LineWidth',2)

for i = 1:length(XT)
    plot(THETA([XT(i) XT(i)+DELTA]), [0 0], 'k-', 'LineWidth',2.5)
end
```

Лістинг функції computeSPL

```
function [ss, ii] = computeSPL(DELTA,THETA,RHO)
ss=[];
ii=[];
for i=DELTA+1:length(THETA)-DELTA
    ql = quad(@(xi) interp1(THETA,RHO,xi), THETA(i-DELTA), THETA(i));
    qr = quad(@(xi) interp1(THETA,RHO,xi), THETA(i), THETA(i+DELTA));
    s = ql-qr;
    fprintf('ql=%.3f qr=%.3f sum=%.3f\n', ql, qr, s)
    ss=[ss s];
    ii=[ii i];
end
```

Лістинг функції порівняння методів виділення характерних точок

```
function [P, E, Q2] = computeQuality(modelPoints, characterPoints, sigma)

characterPoints = characterPoints(1:2,1:end);
diff = zeros(1, size(modelPoints,2));

for j=1:size(modelPoints,2)
    plot(modelPoints(1,j), modelPoints(2,j), 'ks', 'MarkerSize',20,
'LineWidth', 3)
    C = getClosestPoint(characterPoints, modelPoints(:,j));
    diff(j) = sqrt(sum((C-modelPoints(:,j)).^2));
    plot(C(1),C(2),'kh', 'MarkerSize',15, 'LineWidth', 3)

end


Ctotal = size(modelPoints,2)
fprintf('%f,',diff)
Cdetermined = sum(diff <= sigma)
P = Cdetermined / Ctotal
Q = 1-P

% Похибка оцінювання координат
E = sum(diff);
fprintf('\nПохибка оцінювання координат E = %f\n', E)

a=[mean(modelPoints(1,:))-17, mean(modelPoints(2,:))];
```

```
% Ймовірність правильного виявлення
text(a(1),a(2)+10, sprintf('$P=%.3f$', P), 'interpreter','latex', 'FontSize',
28)

% Ймовірність хибного виявлення
text(a(1),a(2), sprintf('$Q=%.3f$', Q), 'interpreter','latex', 'FontSize',
28)

% Похибка оцінювання координат
text(a(1),a(2)-10, sprintf('$E=%.3f$', E), 'interpreter','latex', 'FontSize',
28)

function C = getClosestPoint(characterPoints, P)
D = zeros(1, size(characterPoints,2));

for j=1:size(D,2)
    D(j)=sqrt( sum( (P-characterPoints(:,j)).^2 ));
end

[ans, j] = min(D);
C=characterPoints(:,j);
```

Додаток В

Обчислення ідентифікаційних векторів

Лістинг головної програми

```
function main

close all, clc
I=imread('Prjam_07.JPG');
I = im2bw(I);

[y,x]=find(I);
y=max(y)-y;

mx=(max(x)-min(x))/2;
my=(max(y)-min(y))/2;

x=x-mx;
y=y-my;

figure(1), hold off
plot(x,y,'.')
axis ('image')

[THETA,RHO] = cart2pol(x,y);
[THETA,ii] =sort(THETA);
RHO=RHO(ii);
x=x(ii);
y=y(ii);

DELTA = 8;
[ss, ii] = computeHaar(DELTA,THETA,RHO);

figure(2)

plot(THETA(ii), ss, 'b-', 'LineWidth', 2), hold on %

plot([THETA(ii(1)) THETA(ii(end))], [0 0], 'k-','LineWidth',2)


XT = [];
for i=1:length(ss)-1
    if (ss(i)*ss(i+1)<0) && (isempty(XT) || ii(i)-XT(end)>DELTA)
        [ans, k] = min(abs(ss(i:i+1)))
        if k == 1
            XT=[XT ii(i)];
        else
            XT=[XT ii(i+1)];
        end
    end
end

figure(1), hold on
plot(x(XT), y(XT), 'ro', 'MarkerSize', 10, 'LineWidth',2)%XT вив. в черв круж
XT=[XT XT(1)];
plot(x(XT), y(XT), 'k-', 'LineWidth',2) % сполучає XT

phi = atan2(y(XT), x(XT));
```

```
text(x(XT), y(XT), num2str(phi))

% Обчислення геометричних моментів ознак

% Довжини відрізків:
rho = sqrt(x(XT).^2 + y(XT).^2) % (x,y)  множина точок об'єкту в декартовій
системі координат

fprintf('|phi|=%d, }rho}=%d\n', length(phi), length(rho))

% Площа об'єкта:
S = 0;
for i=1:length(rho)-1
    S=S + 1/2 * rho(i)*rho(i+1) * sin(phi(i+1)-phi(i));
end

fprintf('S=%.3f\n', S)

for p=0:10
    Cp = geometrMomentOznak(rho, S, phi, p);
    fprintf('C%d = %.10f\n', p, Cp)
end

figure(2), hold on
plot(THETA(XT),0,'ro', 'MarkerSize', 10, 'LineWidth',2

for i = 1:length(XT)
    plot(THETA([XT(i) XT(i)+DELTA]), [0 0], 'k-', 'LineWidth',2.5)%
end

function Cp = geometrMomentOznak(rho, S, phi, p)
% rho --- Довжини відрізків
% S --- Площа об'єкта
% phi --- кути XT
% p --- порядок моменту

% Нормований множник:
Z = sqrt(S/2/pi);

Cp = abs(sum( (sqrt(rho(1:end-1).*rho(2:end))/Z ).^(p+2) ...
    .* sin(phi(2:end) - phi(1:end-1)) ));
```

Додаток Г
Структура таблиць розробленої бази даних

Структура таблиці «people»

Назва	Тип даних	Опис
id	Число	Ідентифікатор рядка
fname	Текст	Ім’я
mname	Текст	По-батькові
lname	Текст	Прізвище

Структура таблиці «photos»

Назва	Тип даних	Опис
Id	Число	Ідентифікатор рядка
id_person	Число	Ідентифікатор об’єкта
date	Дата	Дата отримання зображення
filepath	Текст	Шлях до файлу зображення
contour_enabled	Булеве значення	Застосування вейвлет-функцій
wavelet_len	Число	Довжина вейвлету
wavelet_thresh	Число	Поріг
canny_enabled	Булеве значення	Застосування методу на базі методу Канні
canny_thresh1	Число	Поріг 1
canny_thresh2	Число	Поріг 2

Структура таблиці «feature_points»

Назва	Тип даних	Опис
id	Число	Ідентифікатор рядка
id_photo	Число	Ідентифікатор зображення
point_position	Число	Позиція точки
x	Число	Координата x
y	Число	Координата y

Структура таблиці «found_objects»

Назва	Тип даних	Опис
id	Число	Ідентифікатор рядка
media_type	Текст	Тип вхідного файлу
media_seek_time	Число	Час співпадіння з шуканим об’єктом
file_path	Текст	Шлях до вхідного файлу
id_person	Число	Ідентифікатор об’єкта
duration	Число	Крок аналізу кадрів

Додаток Д
Програмна реалізація прикладної інформаційної технології

// Person.cs
// Модель даних особи

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace face_feature_points_detector
{
    public class Person
    {
        public Person() { }
        public Person(string firstName, string middleName, string lastName)
        {
            this.firstName = firstName;
            this.middleName = middleName;
            this.lastName = lastName;
        }

        public Int32 id = 0;
        public string firstName;
        public string middleName;
        public string lastName;

        public override string ToString()
        {
            return String.Format("{0}", lastName);
        }
    }
}
```

// Photo.cs
// Модель даних зображення

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace face_feature_points_detector
{
    public class Photo
    {
        public Int32 id = 0;
        public Int32 idPerson;
        public Int64 date;
        public String filePath;
        public Boolean contourEnabled;
        public Int32 waveletLen;
        public Double waveletThresh;
        public Boolean cannyEnabled;
        public Double cannyThreshold1;
        public Double cannyThreshold2;
```

```
        public override string ToString()
        {
            return String.Format("#{0} {1}", id, filePath);
        }
    }
}
```

//PhotosDao.cs
// Об’єкт доступу до даних зображень

```
using System;
using System.Collections.Generic;
using System.Data.SQLite;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace face_feature_points_detector
{
    class PhotosDao : AbstractDao
    {
        public static Photo add(Photo photo)
        {
            var conn = getConnection();
            var command = conn.CreateCommand();
            command.CommandText = "insert into photos (id_person, date, filepath, 
contour_enabled, wavelet_len, wavelet_thresh, canny_enabled, canny_thresh1, canny_thresh2) 
values (@id_person, @date, @filepath, @contour_enabled, @wavelet_len, @wavelet_thresh, 
@canny_enabled, @canny_thresh1, @canny_thresh2)";
            command.Parameters.Add(new SQLiteParameter("@id_person", photo.idPerson));
            command.Parameters.Add(new SQLiteParameter("@date", photo.date));
            command.Parameters.Add(new SQLiteParameter("@filepath", photo.filePath));
            command.Parameters.Add(new SQLiteParameter("@contour_enabled", 
photo.contourEnabled));
            command.Parameters.Add(new SQLiteParameter("@wavelet_len", photo.waveletLen));
            command.Parameters.Add(new SQLiteParameter("@wavelet_thresh", 
photo.waveletThresh));
            command.Parameters.Add(new SQLiteParameter("@canny_enabled", photo.cannyEnabled));
            command.Parameters.Add(new SQLiteParameter("@canny_thresh1", 
photo.cannyThreshold1));
            command.Parameters.Add(new SQLiteParameter("@canny_thresh2", 
photo.cannyThreshold2));
            command.ExecuteNonQuery();
            var id = (Int32)conn.LastInsertRowId;

            return new Photo
            {
                id = id,
                idPerson = photo.idPerson,
                date = photo.date,
                filePath = photo.filePath,
                contourEnabled = photo.contourEnabled,
                waveletLen = photo.waveletLen,
                waveletThresh = photo.waveletThresh,
                cannyEnabled = photo.cannyEnabled,
                cannyThreshold1 = photo.cannyThreshold1,
                cannyThreshold2 = photo.cannyThreshold2
            };
        }

        public static List<Photo> all()
        {
            var photos = new List<Photo>();
```

```
            var conn = getConnection();

            var command = conn.CreateCommand();
            command.CommandText = "select * from photos";

            var result = command.ExecuteReader();
            while (result.Read())
            {
                photos.Add(new Photo
                {
                    id = Convert.ToInt32(result["id"]),
                    idPerson = Convert.ToInt32(result["id_person"]),
                    date = 0,
                    filePath = Convert.ToString(result["filepath"]),
                    contourEnabled = Convert.ToBoolean(result["contour_enabled"]),
                    waveletLen = Convert.ToInt32(result["wavelet_len"]),
                    waveletThresh = Convert.ToDouble(result["wavelet_thresh"]),
                    cannyEnabled = Convert.ToBoolean(result["canny_enabled"]),
                    cannyThreshold1 = Convert.ToDouble(result["canny_thresh1"]),
                    cannyThreshold2 = Convert.ToDouble(result["canny_thresh2"])
                });
            }

            return photos;

        }
    }
}
```

// PeopleDao.cs
// Обєкт доступу до даних для збереження шуканих осіб (DAO)

```
using System;
using System.Collections.Generic;
using System.Data.SQLite;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace face_feature_points_detector
{
    public class PeopleDao : AbstractDao
    {

        public static Person add(Person person)
        {
            var conn = getConnection();
            var command = conn.CreateCommand();
            command.CommandText = "insert into people (fname, mname, lname) values
(@fname,@mname,@lname)";
            command.Parameters.Add(new SQLiteParameter("@fname", person.firstName));
            command.Parameters.Add(new SQLiteParameter("@mname", person.middleName));
            command.Parameters.Add(new SQLiteParameter("@lname", person.lastName));
            command.ExecuteNonQuery();
            var id = (Int32)conn.LastInsertRowId;

            return new Person
            {
                id = id,
                firstName = person.firstName,
```

```
                middleName = person.middleName,
                lastName = person.lastName
            };
        }

        public static Person findByName(Person pattern)
        {
            var conn = getConnection();

            var command = conn.CreateCommand();
            command.CommandText = "select id,fname,mname,lname from people where fname=@fname
and mname=@mname and lname=@lname limit 1";
            command.Parameters.Add(new SQLiteParameter("@fname", pattern.firstName));
            command.Parameters.Add(new SQLiteParameter("@mname", pattern.middleName));
            command.Parameters.Add(new SQLiteParameter("@lname", pattern.lastName));
            var result = command.ExecuteReader();
            if (result.Read())
            {
                return toPerson(result);
            }
            else
            {
                return null;
            }

        }

        private static Person toPerson(SQLiteDataReader result)
        {
            return new Person
            {
                id = Convert.ToInt32(result["id"]),
                firstName = Convert.ToString(result["fname"]),
                middleName = Convert.ToString(result["mname"]),
                lastName = Convert.ToString(result["lname"])
            };
        }

        public static List<Person> all()
        {
            var points = new List<Person>();
            var conn = getConnection();

            var command = conn.CreateCommand();
            command.CommandText = "select * from people";
            var result = command.ExecuteReader();
            while (result.Read())
            {
                points.Add(toPerson(result));
            }
            return points;

        }
    }
}
```

// AddFaceForm.cs
// Екранна форма виявлення ХТ контуру відеозображень

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Diagnostics;
//using OpenCvSharp;

namespace face_feature_points_detector
{
    public partial class AddFaceForm : Form, AddFaceView
    {
        private OpenFileDialog openFileDialog = new OpenFileDialog();
        AddFacePresenter addFacePresenter;

        public string FirstName
        {
            get
            {
                return textBoxFirstName.Text;
            }
        }

        public string MiddleName
        {
            get
            {
                return textBoxMiddleName.Text;
            }
        }

        public string LastName
        {
            get
            {
                return textBoxLastName.Text;
            }
        }

        public bool isContourEnabled
        {
            get
            {
                return checkBoxFaceContour.Checked;
            }
        }

        public int waveletLength
        {
            get
            {
                return trackBarWaveletLength.Value;
            }
        }

        public double waveletThreshold
        {
```

```
    get
    {
        return trackBarThreshold.Value;
    }
}

public bool isCannyEnabled
{
    get
    {
        return checkBoxCanny.Checked;
    }
}

public double cannyThresh1
{
    get
    {
        return trackBarCannyThreshold1.Value;
    }
}

public double cannyThresh2
{
    get
    {
        return trackBarCannyThreshold2.Value;
    }
}

public bool checkBoxFaceChecked
{
    get
    {
        return checkBoxFace.Checked;
    }
}

public bool checkBoxCannyChecked
{
    get
    {
        return checkBoxCanny.Checked;
    }
}

public bool checkBoxApplyWaveletChecked
{
    get
    {
        return checkBoxApplyWavelet.Checked;
    }
}

public bool checkBoxFaceContourChecked
{
    get
    {
        return checkBoxFaceContour.Checked;
    }
}

public bool checkBoxMouthChecked
{
    get
```

```
            {
                return checkBoxMouth.Checked;
            }
        }

        public bool checkBoxBrowesChecked
        {
            get
            {
                return checkBoxBrowes.Checked;
            }
        }

        public AddFaceForm(int number, Image image,List<Point> face)
        {
            addFacePresenter = new AddFacePresenter(this, face, image);
            InitializeComponent();
            this.textBoxNumber.Text = "#" + number;
            pictureBox1.Image = image;
        }

        private void label2_Click(object sender, EventArgs e)
        {

        }

        private void addFaceButtonClick(object sender, EventArgs e)
        {
            addFacePresenter.addFeaturePoints();
            Hide();
        }

        private void checkBoxFaceCheckedChanged(object sender, EventArgs e)
        {
            updateImage();
        }

        void updateImage() {
            textBoxThreshold.Text = "" + trackBarThreshold.Value;
            var waveletLength = trackBarWaveletLength.Value;
            textBoxWaveletLength.Text = String.Format("{0}", waveletLength);
            textBoxCannyThreshold1.Text = "" + trackBarCannyThreshold1.Value;
            textBoxCannyThreshold2.Text = "" + trackBarCannyThreshold2.Value;

            pictureBox1.Image = addFacePresenter.updateImage();
        }

        private void trackBarWaveletLength_Scroll(object sender, EventArgs e)
        {
            updateImage();
        }
        private void checkBoxApplyWavelet_CheckedChanged(object sender, EventArgs e)
        {
            updateImage();
        }
        private void trackBarThreshold_Scroll(object sender, EventArgs e)
        {
            updateImage();
        }
        private void label5_Click(object sender, EventArgs e)
        {
        }
    }
}
```

// MainForm.cs
// Головна екранна форма застосунку

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Diagnostics;
using System.Threading;

namespace face_feature_points_detector
{
    public partial class MainForm : Form, ContourDetectorView
    {
        OpenFileDialog openFileDialog = new OpenFileDialog();

        ContourDetectorPresenter contourDetectorService;

        public FeaturePointComparationType featurePointComparationType
        {
            get
            {
                return класичнийЕвклідаToolStripMenuItem.Checked ?
FeaturePointComparationType.StandardEuclidian :
                    FeaturePointComparationType.ModifiedEuclidian;
            }
        }

        public ClassificationMethodType classificationMethodType
        {
            get
            {
                if (центроїдToolStripMenuItem.Checked)
                {
                    return ClassificationMethodType.CentroidClassification;
                }
                if (максимумToolStripMenuItem.Checked)
                {
                    return ClassificationMethodType.MaxClassification;
                }
                return ClassificationMethodType.SumClassification;
            }
        }

        public MainForm()
        {
            contourDetectorService = new ContourDetectorPresenter(this);
            changeUiLang("uk");
        }

        private void buttonAddClick(object sender, EventArgs e)
        {
            //listBoxFaces.Items.Add( "" + FaceHandler.Add(1, 2));
            contourDetectorService.addContour(listBoxFaces.SelectedIndex);
        }

        private void відкритиToolStripMenuItem_Click(object sender, EventArgs e)
        {
            var res = openFileDialog.ShowDialog();
            if (res == DialogResult.OK)
```

```
            {
                //Pash - uncom
                //pictureBox.ImageLocation = openFileDialog.FileName;

                contourDetectorService.picture = Image.FromFile(openFileDialog.FileName);

                contourDetectorService.detectContour(openFileDialog.FileName);
            }
        }

        private void базаОбєктівToolStripMenuItem_Click(object sender, EventArgs e)
        {
            new FaceViewerForm().Show();
        }

        void drawFace(Graphics g, List<Point> face)
        {
            foreach(var p in face)
            {
                g.DrawEllipse(new Pen(Brushes.Red, 3), p.X, p.Y, 3, 3);
            }
        }

        private void українськаToolStripMenuItem_Click(object sender, EventArgs e)
        {
            changeUiLang("uk");
        }

        private void changeUiLang(string lang)
        {
            Thread.CurrentThread.CurrentUICulture = new
System.Globalization.CultureInfo(lang);
            Controls.Clear();
            InitializeComponent();
            класичнийЕвклідаToolStripMenuItem.Checked = true;
        }

        private void englishToolStripMenuItem_Click(object sender, EventArgs e)
        {
            changeUiLang("en");
        }

        private void налаштуванняToolStripMenuItem_Click(object sender, EventArgs e)
        {

        }

        private void listBoxFaces_SelectedIndexChanged(object sender, EventArgs e)
        {
            var number = listBoxFaces.SelectedIndex;
            contourDetectorService.setCurrentFace(number);
            contourDetectorService.showDetectedPeople();
        }

        public void showDetectedPeople(List<DetectedPerson> people)
        {
            listBoxSimilarity.Items.Clear();
            foreach (var detectedPerson in people)
            {
                listBoxSimilarity.Items.Add(detectedPerson);
            }
        }

        public void addFaceIno(string faceInfo)
        {
```

```
            listBoxFaces.Items.Add(faceInfo);
        }

        public void showPicture(Image picture)
        {
            pictureBox.Image = picture;
        }

        private void listBoxSimilarity_SelectedIndexChanged(object sender, EventArgs e)
        {
            var detectedPerson = listBoxSimilarity.SelectedItem as DetectedPerson;
            pictureBoxDetected.ImageLocation =
ImageUtils.filePath(detectedPerson.photo.filePath);
        }

        private void класичнийЕвклідаToolStripMenuItem_Click(object sender, EventArgs e)
        {
            модифікованийЕвклідаToolStripMenuItem.Checked = false;
            класичнийЕвклідаToolStripMenuItem.Checked = true;
            contourDetectorService.updateFaceInfo();
        }

        private void модифікованийЕвклідаToolStripMenuItem_Click(object sender, EventArgs e)
        {
            модифікованийЕвклідаToolStripMenuItem.Checked = true;
            класичнийЕвклідаToolStripMenuItem.Checked = false;
            contourDetectorService.updateFaceInfo();
        }

        public void resetFaceInfo()
        {
            listBoxFaces.Items.Clear();
        }

        private void центроїдToolStripMenuItem_Click(object sender, EventArgs e)
        {
            центроїдToolStripMenuItem.Checked = true;
            сумаToolStripMenuItem.Checked = false;
            максимумToolStripMenuItem.Checked = false;
            contourDetectorService.updateFaceInfo();
        }

        private void сумаToolStripMenuItem_Click(object sender, EventArgs e)
        {
            центроїдToolStripMenuItem.Checked = false;
            сумаToolStripMenuItem.Checked = true;
            максимумToolStripMenuItem.Checked = false;
            contourDetectorService.updateFaceInfo();
        }

        private void максимумToolStripMenuItem_Click(object sender, EventArgs e)
        {
            центроїдToolStripMenuItem.Checked = false;
            сумаToolStripMenuItem.Checked = false;
            максимумToolStripMenuItem.Checked = true;
            contourDetectorService.updateFaceInfo();
        }
    }
}
```

// FacesService.cs
// Сервіс обробки відеоінформації

```
using System;
using System.Collections.Generic;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace face_feature_points_detector
{
    public class FacesService
    {
        List<FeaturePoint> featurePoints;
        List<Person> people;
        List<Photo> photos;
        FeaturePointComparationType featurePointComparationType;
        ClassificationMethodType classificationMethodType;

        public FacesService(FeaturePointComparationType featurePointComparationType,
            ClassificationMethodType classificationMethodType)
        {
            this.classificationMethodType = classificationMethodType;
            this.featurePointComparationType = featurePointComparationType;
            featurePoints = FeaturePointsDao.all();
            people = PeopleDao.all();
            photos = PhotosDao.all();
        }

        public List<DetectedPerson> detectPerson(List<Point> newFacePoints)
        {
            var newFaceFeaturePoints = ImageUtils.toFeaturePoints(newFacePoints, 0);
            var detectedPeople = new List<DetectedPerson>();
            foreach (var person in people)
            {
                switch (classificationMethodType)
                {
                    case ClassificationMethodType.CentroidClassification:
                        detectedPeople.Add(measureSimilarityCentroid(newFaceFeaturePoints,
person));
                        break;
                    case ClassificationMethodType.MaxClassification:
                        detectedPeople.Add(measureSimilarityMax(newFaceFeaturePoints,
person));
                        break;
                    case ClassificationMethodType.SumClassification:
                        detectedPeople.Add(measureSimilaritySum(newFaceFeaturePoints,
person));
                        break;
                }

            }
            return detectedPeople.OrderBy(x=> x.percentOfSimilarity).Reverse().ToList();
        }

        private DetectedPerson measureSimilaritySum(List<FeaturePoint> newFacePoints, Person
person)
        {
            var personPhotos = photos.FindAll(x => x.idPerson == person.id);
            var personSimilarity = 1 / ( 1 +
                personPhotos.Select(photo => measureDistance(newFacePoints, photo)).Sum());

            return new DetectedPerson {
```

```
                person = person,
                percentOfSimilarity = personSimilarity,
                photo = photos.First()
            };
        }

        private DetectedPerson measureSimilarityCentroid(List<FeaturePoint> newFacePoints,
Person person)
        {
            var personPhotos = photos.FindAll(x => x.idPerson == person.id);
            var personSimilarity = 1 / (1 + measureDistance(newFacePoints, centroid(person)));

            return new DetectedPerson
            {
                person = person,
                percentOfSimilarity = personSimilarity,
                photo = photos.First()
            };
        }

        private DetectedPerson measureSimilarityMax(List<FeaturePoint> newFacePoints, Person
person)
        {
            var personPhotos = photos.FindAll(x => x.idPerson == person.id);
            var personWithSimilarity = personPhotos.Select(photo => new DetectedPerson
            {
                person = person,
                photo = photo,
                percentOfSimilarity = 1 / (1 + measureDistance(newFacePoints, photo))
            }).OrderBy(x => x.percentOfSimilarity).LastOrDefault();

            return personWithSimilarity ?? new DetectedPerson { person = person,
percentOfSimilarity = 0 };
        }

        List<FeaturePoint> centroid(Person person)
        {
            var personPhotos = photos.FindAll(x => x.idPerson == person.id);
            var summed = centroid(personPhotos[0]);
            for (var i=1; i<personPhotos.Count; ++i)
            {
                var next = centroid(personPhotos[i]);
                for (var j=0; j<summed.Count; j++)
                {
                    summed[j].x += next[j].x;
                    summed[j].y += next[j].y;
                }
            }
            for (var j = 0; j < summed.Count; j++) {
                summed[j].x /= personPhotos.Count;
                summed[j].y /= personPhotos.Count;
            }
            return summed;
        }

        List<FeaturePoint> centroid(Photo photo)
        {
            return featurePoints.FindAll(x => x.idPhoto == photo.id)
                .OrderBy(x => x.pointPosition).ToList();
        }

        private Double measureDistance(List<FeaturePoint> newFaceFeaturePoints, Photo photo)
        {
            var photoFeaturePoints = featurePoints.FindAll(x => x.idPhoto == photo.id)
                .OrderBy(x => x.pointPosition).ToList();
```

```
            return measureDistance(newFaceFeaturePoints, photoFeaturePoints);
        }

        private double measureDistance(List<FeaturePoint> newFaceFeaturePoints,
List<FeaturePoint> photoFeaturePoints)
        {
            if (photoFeaturePoints.Count == 0)
            {
                return Double.MaxValue;
            }
            var N = Math.Min(photoFeaturePoints.Count, newFaceFeaturePoints.Count);
            var sumDiff = 0.0;
            for (var i = 0; i < N; ++i)
            {
                if (featurePointComparationType ==
FeaturePointComparationType.StandardEuclidian)
                {
                    sumDiff += sqr(photoFeaturePoints[i].x - newFaceFeaturePoints[i].x) +
                        sqr(photoFeaturePoints[i].y - newFaceFeaturePoints[i].y);
                }
                else
                {
                    sumDiff += sqr(sqr(photoFeaturePoints[i].x) + sqr(photoFeaturePoints[i].y)
-
                        sqr(newFaceFeaturePoints[i].x) - sqr(newFaceFeaturePoints[i].y));
                }
            }
            return Math.Sqrt(sumDiff);
        }

        static Double sqr(Double x)
        {
            return x * x;
        }
    }
}
```

// ContourDetectorPresenter.cs
// Презентаційний компонент виявлення контуру

```
using System;
using System.Collections.Generic;
using System.Diagnostics;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace face_feature_points_detector
{
    public class ContourDetectorPresenter
    {
        const double PersonDetectionPercentThreshold = 0.7;
        public ContourDetectorPresenter(ContourDetectorView contourDetectorView)
        {
            this.contourDetectorView = contourDetectorView;
        }

        List<List<Point>> faces;
        List<Point> face;
```

```
public Image picture;
const float coefY = 0.01f;
const float coefX = 0.01f;
private Point imageSize;
private ContourDetectorView contourDetectorView;
FacesService facesService;

public void setCurrentFace(int faceIndex)
{
    if (faceIndex >= 0)
    {
        face = faces[faceIndex];
        drawFaceRectangles(faceIndex + 1);
    }
}

public void addContour(int selectedFace)
{
    var cropRect = getFaceRect(face);

    Bitmap target = new Bitmap(cropRect.Width, cropRect.Height);
    using (Graphics g = Graphics.FromImage(target))
    {
        g.DrawImage(picture, new Rectangle(0, 0, target.Width, target.Height),
                         cropRect,
                         GraphicsUnit.Pixel);
    }
    var shiftedFace = new List<Point>();
    foreach (var p in face)
        shiftedFace.Add(new Point(p.X - cropRect.X, p.Y - cropRect.Y));

    new AddFaceForm(selectedFace + 1, target, shiftedFace).Show();
}

public void showDetectedPeople()
{
    facesService = new FacesService(contourDetectorView.featurePointComparationType,
        contourDetectorView.classificationMethodType);
    var detectedPeople = facesService.detectPerson(face);
    contourDetectorView.showDetectedPeople(detectedPeople);
}

public void detectContour(string filepath)
{
    Process.Start(@"face_landmark_detection_ex.exe", filepath).WaitForExit();
    var file = System.IO.File.ReadAllLines(filepath + ".points.out");

    this.imageSize = parsePoint(file[0]);

    faces = new List<List<Point>>();

    parseFacePoints(1, file, faces, imageSize);

    drawFaceRectangles(-1);

    updateFaceInfo();
}

public void updateFaceInfo()
{
    facesService = new FacesService(contourDetectorView.featurePointComparationType,
        contourDetectorView.classificationMethodType);
    var number = 1;
    contourDetectorView.resetFaceInfo();
    foreach (var face in faces)
```

```
            {
                var mostSimilarPerson = facesService.detectPerson(face).First();
                var objectName =
                    (mostSimilarPerson.percentOfSimilarity > PersonDetectionPercentThreshold)
?
                        mostSimilarPerson.ToString() : "Невідомий об`єкт";

                contourDetectorView.addFaceIno(String.Format(" #{0} - {1}", number++,
objectName));
            }
        }

        private void drawFaceRectangles(int selectedFaceIndex)
        {
            var pictureWithRectangles = picture.Clone() as Image;
            var g = Graphics.FromImage(pictureWithRectangles);
            //g.DrawEllipse(new Pen(Brushes.Red), 100, 100, 100, 100);
            var number = 1;
            foreach (var face in faces)
            {
                drawFaceRectangle(g, face, number, number == selectedFaceIndex ?
                     Brushes.Red : Brushes.LightGreen);
                number++;
            }
            g.Dispose();
            contourDetectorView.showPicture(pictureWithRectangles);
        }

        Point parsePoint(string str)
        {
            var line = str.Split("(, )".ToCharArray());
            var x = Int32.Parse(line[1]);
            var y = Int32.Parse(line[3]);
            return new Point(x, y);
        }

        public void parseFacePoints(int start, string[] points, List<List<Point>> faces, Point
imageSize)
        {
            if (start >= points.Length)
            {
                MessageBox.Show(String.Format("start {0} points {1}", start, points));
                return;
            }
            SizeF si = picture.Size;
            var face = new List<Point>();
            var len = Int32.Parse(points[start]);
            for (int i = 1; i <= len; i++)
            {
                var point = parsePoint(points[i + start]);
                face.Add(new Point((int)(point.X / (float)imageSize.Y * si.Width),
                    (int)(point.Y / (float)imageSize.X * si.Height)));
            }
            faces.Add(face);
            var next = len + start + 2;
            if (next >= points.Length)
            {
                return;
            }
            parseFacePoints(next, points, faces, imageSize);
        }


        Rectangle getFaceRect(List<Point> face)
        {
```

```
            var xAdd = Math.Max(picture.Width, picture.Height) * coefX;
            var yAdd = xAdd;
            var top = 2f * face[30].Y - face[8].Y - 3.5f * yAdd;
            var height = face[8].Y - top + yAdd * 1.5f;
            var left = face[0].X - xAdd;
            var width = face[16].X - face[0].X + xAdd * 3; //*2
            return new Rectangle((int)left, (int)top, (int)width, (int)height);
        }

        void drawFaceRectangle(Graphics g, List<Point> face, int number, Brush brush)
        {
            var fontScale = Math.Max(imageSize.X, imageSize.Y) * .02f; // .01f
            var rect = getFaceRect(face);
            g.DrawRectangle(new Pen(brush, 2),
                rect);
            Font drawFont = new Font("Arial", fontScale);
            g.DrawString(String.Format("#{0}", number), drawFont, Brushes.LightGreen,
                new PointF(rect.X, rect.Y - 1.5f * fontScale));

        }

    }
}
```

// ContourDetectorView.cs
// Абстрактний компонент відображення контурів

```
using System;
using System.Collections.Generic;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace face_feature_points_detector
{
    public interface ContourDetectorView
    {
        FeaturePointComparationType featurePointComparationType { get; }
        ClassificationMethodType classificationMethodType { get; }

        void showDetectedPeople(List<DetectedPerson> people);
        void resetFaceInfo();
        void addFaceIno(string faceInfo);
        void showPicture(Image picture);
    }
}
```

Додаток Е

Наукові праці здобувача

Статті в наукових фахових виданнях та закордонних журналах

1. Загородня Д.І. Дослідження диференціального методу виділення характерних точок контуру облич / Д.І. Загородня. В.І. Дорош, Н.С. Добровольська, О.Л. Римар // Міжнародний науково-технічний журнал «Вимірювальна та обчислювальна техніка в технологічних процесах». – Хмельницький, 2014. – №4 (49). – С. 162-166.

2. Загородняя Д.И. Метод идентификации лиц по характерным точкам контура / Д.И. Загородняя // Вестник Брестского государственного технического университета. Серия физика, математика, информатика. – 2015. – №5. – С. 30-33.

3. Загородня Д.І. Аналіз інтерполяційного методу виділення характерних точок контуру / Д.І. Загородня // Науковий вісник Чернівецького національного університету ім. Ю. Федьковича. Серія: Комп'ютерні системи та компоненти. – 2015. – Том 6, Випуск 2. – С. 88-92.

4. Загородня Д.І. Порівняння методів виділення характерних точок контуру / Д.І. Загородня // Радіоелектронні і комп'ютерні системи. Харків «ХАІ». – 2016. – № 3 (77). – С. 56-61.

5. Zahorodnia D.I. Canny-based Method of Image Contour Segmentation / Diana Zahorodnia, Yuriy Pigovsky, Pavlo Bykovyy // International Journal of Computing. Ternopil: TNEU. – 2016. – Vol. 15. – Issue 3. – P. 200-205. (*включено до науко-метричної бази Scopus*).

6. Загородня Д.І. Метод виділення контурів зображень з використанням вейвлет-перетворень / Д.І. Загородня, У. Гросман, В.М. Крилов, В.І. Дорош // Науковий вісник Чернівецького національного університету ім. Ю. Федьковича. Серія: Комп'ютерні системи та компоненти. –2016. – Том 7, Випуск 2. – С. 136-141.

Опубліковані праці апробаційного характеру

7. Paliy Ihor. Fast and Robust Face Detection and Tracking Framework / Ihor Paliy, Volodymyr Dovgan, Ognian Boumbarov, Stanislav Panev, Anatoly Sachenko, Yuriy Kurylyak, Diana Zagorodnya // Proceedings of the IEEE 6th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS'2011). – Prague (Czech Republic), September 15-17, 2011. – Vol.1. – P. 430-433. (*включено до науко-метричної бази Scopus*)

8. Zahorodnia Diana. Structural Statistic Method Identifying Facial Images by Countour Characteristic Points / Diana Zahorodnia, Yuriy Pigovsky, Pavlo Bykovyy, Viktor Krylov, Ihor Paliy, Igor Dobrotvor // Proceedings of the 8th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS'2015). – Warsaw (Poland), September 24-26, 2015. – P. 293-297. (*включено до науко-метричної бази Scopus та Web of Science*)

9. Базелюк Д.В. Підхід до розпізнавання емоцій на основі зображення обличчя. / Д.В. Базелюк, Д.І. Загородня, І.О. Палій // Труди XIII міжнародної науково-практичної конференції «Сучасні інформаційні та електронні технології» («СІЕТ-2012»). – Одеса, 4-8 червня 2012. – С. 118.

10. Загородня Д.І. Алгоритми прослідковування контурів та виділення характерних точок в процедурі сегментації системи комп'ютерного розпізнавання / Д.І. Загородня, І.О. Палій, В.М. Крилов // Сучасні комп'ютерні інформаційні технології: Матеріали ІІІ Всеукраїнської школи-семінару молодих вчених і студентів (ACIT'2013). – Тернопіль: ТНЕУ, 17-18 травня 2013. – С.129-130.

11. Загородня Д.І. Метод розпізнавання облич за характерними точками контуру в комп'ютерних системах відеоспостереження. / Д.І. Загородня, І.О. Палій, В.М. Крилов // Труди XIV міжнародної науково-практичної конференції «Сучасні інформаційні та електронні технології» («СІЕТ-2013»). – Одеса, 27-31 травня 2013. – С. 49-52.

12. Загородня Д.І. Алгоритм контурної сегментації в системах відеоспостереження / Д.І. Загородня, К.Ю. Ковалок, А.О. Саченко, В.М. Крилов // Всеукраїнська наукова Інтернет-конференція «Економічна кібернетика –

інженерія економіки (випуск 24)» / Збірник тез доповідей: випуск 24. – Тернопіль: Тайп, 29-30 листопада 2013. – С. 3-5.

13. Zahorodnia Diana. Contour Segmentation Method in Video Surveillance Systems / Diana Zahorodnia, Kostantin Kovalok, Anatoly Sachenko, Viktor Krylov, Sergiy Nychyporuk // Proceedings of the International Conference "Modern Problems of Radio Engineering, Telecommunications and Computer Science" (TCSET'2014). – Lviv-Slavske (Ukraine), February 25 – March 1, 2014. – P. 405.

14. Загородня Д.І. Налаштування параметрів алгоритму виділення характерних точок на основі вейвлету Хаара / Д.І. Загородня // Сучасні комп'ютерні інформаційні технології: матеріали V Всеукраїнської школи-семінару молодих вчених і студентів ACIT'2015. – Тернопіль: ТНЕУ, 22-23 травня 2015. – С. 68-70.

15. Загородня Д.І. Метод виділення характерних точок зображення на основі вейвлет-аналізу функції кривизни / Д.І. Загородня, К.Ю. Ковалок, В.М. Крилов // Труди XVI міжнародної науково-практичної конференції «Сучасні інформаційні та електронні технології» («СІЕТ-2015») – Одеса, 25-29 травня 2015. – С. 50-51.

16. Загородня Д.І. Підхід до структурно-ієрархічної контурної сегментації / Д.І. Загородня // Інформатика, управління та штучний інтелект. Матеріали другої науково-технічної конференції студентів, магістрів та аспірантів. – Харків: НТУ «ХПІ», 26-27 листопада 2015. – С. 31.

17. Zahorodnia D.I. Structural-Hierarchical Principle of Contour Segmentation / D.I. Zahorodnia, P.E. Bykovyy // Current Issues in Modern Technologies: Book of abstracts of the IV International Scientific and Technical Conference of Young Researchers and Students. – Ternopil: TNTU, 25-26 November 2015. – Vol. 2. – P. 20-21.

18. Загородня Д.І. Контроль доступу до комп'ютера на основі розпізнавання зображення обличчя користувача / Д.І. Загородня, О.І. Губчакевич, В.В. Кисіль. // Актуальні задачі сучасних технологій: збірник тез доповідей V міжнародної науково-технічної конференції молодих учених та студентів,– Тернопіль: ТНТУ, 17–18 листопада 2016. – Том 2. – С. 42-43.

Додаток Ж

Опубліковані праці апробаційного характеру

№ п/п	Назви конференції, конгресу, симпозіуму, семінару, школи	Місце проведення	Дата проведення (із зазначенням числа, місяця та року)	Форма участі
1.	6th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS'2011)	м. Прага, Чеська республіка	15-17 вересня 2011 року	Заочна
2.	Тринадцята міжнародна науково-практична конференціях "Сучасні інформаційні та електронні технології" («СИЭТ-2012»)	м. Одеса	4-8 червня 2012 року	Очна
3.	ІІІ Всеукраїнська школа-семінар молодих вчених і студентів «Сучасні комп'ютерні інформаційні технології» (ACIT-2013).	м. Тернопіль	17-18 травня 2013 року	Очна
4.	Чотирнадцята міжнародна науково-практична конференціях "Сучасні інформаційні та електронні технології" («СИЭТ-2013»)	м. Одеса	27-31 травня 2013 року	Очна
5.	Всеукраїнська наукова Інтернет-конференція «Економічна кібернетика – інженерія економіки (випуск 24)»	м. Тернопіль	29-30 листопада 2013 року	Заочна
6.	International Conference "Modern Problems of Radio Engineering, Telecommunications and Computer Science" (TCSET'2014)	м. Львів-Славське	25 лютого – 1 березня 2014 року	Заочна
7.	V Всеукраїнська школа-семінар молодих вчених і студентів «Сучасні комп'ютерні інформаційні технології» (ACIT-2015).	м. Тернопіль	22-23 травня 2015 року	Очна
8.	Шістнадцята міжнародна науково-практична конференціях "Сучасні інформаційні та електронні технології" («СИЭТ-2015»)	м. Одеса	25-29 травня 2015 року	Очна
9.	8th IEEE International	м. Варшава,	24-26 вересня 2015	Очна

	Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS'2015).	Республіка Польща	року	
10.	Науково-технічна конференція студентів, магістрів та аспірантів «Інформатика, управління та штучний інтелект» («ИУИИ-2015»)	м. Харків	26-27 листопада 2015 року	заочна
11.	IV International Scientific and Technical Conference of Young Researchers and Students «Current Issues in Modern Technologies»	м. Тернопіль	25-26 листопада 2015 року	очна
12.	V міжнародна науково-технічна конференція молодих учених та студентів «Актуальні задачі сучасних технологій»	м. Тернопіль	17-18 листопада 2016 року	очна

Додаток И

Акти впровадження результатів дисертаційної роботи

ECODERY

46009 Тернопільська область
м. Тернопіль
вул. Гетьмана І.Мазепи, будинок 12
тел.(0352)400970, www.ecodery.com
р/р 26005556114400 АТ"УкрСиббанк" м.Харків МФО 351005
ЄДРПОУ 39636962 і.п.н.396369619183
платник єдиного податку (3-я група)

12 Mazepy str, Ternopil, Ukraine, 46009
Phone+3(80352)400970, www.ecodery.com
acc.№ 26005556114400 UKRSIBBANK, Charkiv City, MFO code 351005
State registration number 39636962

ПРИВАТНЕ ПІДПРИЄМСТВО «ЕКОДЕРІ»

АКТ

про впровадження результатів дисертаційної роботи аспірантки кафедри інформаційно-обчислювальних систем і управління Тернопільського національного економічного університету

Загородньої Діани Іванівни на тему "Інформаційна технологія структурно-статистичної ідентифікації ієрархічних об'єктів за характерними точками їх контурів в автоматизованих системах відеонагляду", представленої на здобуття наукового ступеня кандидата технічних наук

Даний акт складений у тому, результати дисертаційної роботи Загородньої Д.І. на тему «Інформаційна технологія структурно-статистичної ідентифікації ієрархічних об'єктів за характерними точками їх контурів в автоматизованих системах відеонагляду» впроваджені при розробці програмного модуля автоматизованої системи відеонагляду для контурної сегментації зображень на базі методу Канні та базової інформаційної технології контурної сегментації, відмінною особливістю яких є використання вейвлет-функцій в якості підкреслюючого перетворення.

Даний модуль дозволив отримувати послідовність контурних препаратів ієрархічного об'єкта з регульованою деталізацією, що дозволило підвищити швидкодію виявлення об'єктів автоматизованою системою відеонагляду.

25.11.2016 р.

Директор **Сентик Н.П.**

Адреса вул. Танцорова 22, м.Тернопіль, Україна, 46002 **ТОВ "ВІП комп"ютер"**
ЄДРПОУ 34419514 МФО 300023
Р/р 26009000258764 в ПАТ" Укрсоцбанк"
ІПН 344195119185, номер свідоцтва 100206043
Є платником податку на прибуток на загальних підставах

VIP КОМП'ЮТЕРИ

42-80-88 Сервісний центр **43-06-64** Офіс, магазин

АКТ

про впровадження результатів дисертаційної роботи Загородньої Діани Іванівни на тему "Інформаційна технологія структурно-статистичної ідентифікації ієрархічних об'єктів за характерними точками їх контурів в автоматизованих системах відеонагляду ", представленої на здобуття наукового ступеня кандидата технічних наук

Результати дисертаційної роботи Загородньої Д.І. на тему «Інформаційна технологія структурно-статистичної ідентифікації ієрархічних об'єктів за характерними точками їх контурів в автоматизованих системах відеонагляду» впроваджені в програмному модулі автоматизованої системи відеонагляду, що реалізує метод та базову інформаційну технологію виділення характерних точок контуру об'єктів з регулюючою деталізацією та базується на вейвлет-аналізі функції кривизни об'єктів.

Даний модуль зменшив кількість характерних точок контуру об'єктів, що дозволило підвищити швидкодію виявлення об'єктів системою відеонагляду.

23.11.2016 р.

Директор : Кіцак Г.В.

ТОВАРИСТВО З ОБМЕЖЕНОЮ ВІДПОВІДАЛЬНІСТЮ "ВІП КОМП'ЮТЕР" VIP КОМП'ЮТЕР 34419514 УКРАЇНА ТЕРНОПІЛЬ

ЗАВОД
РЕМПОБУТТЕХНІКА
товариство з обмеженою відповідальністю

АКТ

про впровадження результатів дисертаційної роботи аспірантки кафедри інформаційно-обчислювальних систем і управління Тернопільського національного економічного університету Загородньої Діани Іванівни на тему "Інформаційна технологія структурно-статистичної ідентифікації ієрархічних об'єктів за характерними точками їх контурів в автоматизованих системах відеонагляду", представленої на здобуття наукового ступеня кандидата технічних наук

Даний акт складений у тому, що результати роботи Загородньої Д.І. на тему «Інформаційна технологія структурно-статистичної ідентифікації ієрархічних об'єктів за характерними точками їх контурів в автоматизованих системах відеонагляду» впроваджені в системі відеонагляду окремим програмним модулем структурно-статистичної ідентифікації об'єктів.

Для виявлення об'єктів даний модуль використовує ієрархічну послідовність ідентифікаційних векторів, розрахованих на основі характерних точок контуру об'єкту з регульованою деталізацією та базову інформаційну технологію структурно-статистичної ідентифікації об'єктів, що дало можливість підвищувати швидкодію виявлення об'єктів системою відеонагляду при високій достовірності.

21.11.2016 р. Директор Л. П. Слободянюк

«ЗАТВЕРДЖУЮ»
Перший проректор
Тернопільського національного
економічного університету
________________ Шинкарик М.І.
«17» 11 __________ 2016 р.

АКТ

про впровадження у навчальний процес Тернопільського національного економічного університету (ТНЕУ) результатів дисертаційної роботи аспірантки кафедри інформаційно-обчислювальних систем і управління Загородньої Діани Іванівни "Інформаційна технологія структурно-статистичної ідентифікації ієрархічних об'єктів за характерними точками їх контурів в автоматизованих системах відеонагляду", що отримані в рамках виконання науково-дослідної роботи "Виділення контурів для структурно-статистичної ідентифікації об'єктів " (державний реєстраційний № 0112U008453)

Ми, комісія у складі: завідувача кафедри інформаційно-обчислювальних систем і управління, д.т.н., проф. Саченка А.О., професора кафедри інформаційно-обчислювальних систем і управління, д.т.н., доц. Яцківа В.В. та доцента кафедри інформаційно-обчислювальних систем і управління, к.т.н. Комара М.П. склали цей акт про те, що в навчальному процесі факультету комп'ютерних інформаційних технологій ТНЕУ для студентів напряму підготовки "Комп'ютерні науки" впроваджені та використовуються наступні результати дисертаційної роботи Загородньої Д.І. при викладанні дисциплін:

1. Комп'ютерна графіка:
   - Робота з контурами простих геметричних фігур;
   - Реалізація методу прослідковування контуру.
2. Теорія алгоритмів:
   - Розробка алгоритму проходження контура двійкового зображення;
   - Розробка алгоритму виділення контурів об'єкту.
3. Людино-комп'ютерна взаємодія:
   - Розробка інтефейсу підсистеми вводу зображень;
   - Розробка інтефейсу вікон виводу графічних результатів.

Ефект від використання результатів дисертаційної роботи Загородньої Д.І. полягає у підвищенні якості вивчення майбутніми фахівцями сучасних методів та алгоритмів обробки зображень, покращенні якості курсового та дипломного проектування, що в результаті забезпечує підвищення рівня підготовки фахівців з напряму "Комп'ютерні науки".

Завідувач кафедри
інформаційно-обчислювальних
систем і управління, д.т.н., професор — Саченко А.О.

Професор кафедри
інформаційно-обчислювальних
систем і управління, д.т.н., доцент — Яцків В.В.

Доцент кафедри
інформаційно-обчислювальних
систем і управління, к.т.н. — Комар М.П.

«ЗАТВЕРДЖУЮ»
Проректор з наукової роботи
Тернопільського національного
економічного університету
проф. Задорожний З.-М. В.

«____» __________ 2016 р.

АКТ
про впровадження результатів дисертаційної роботи аспірантки кафедри інформаційно-обчислювальних систем і управління (ІОСУ) Загородньої Діани Іванівни
у науково-дослідній роботі на тему:
«Виділення контурів для структурно-статистичної ідентифікації об'єктів»
(державний реєстраційний № 0112U008453)

Ми, комісія у складі директора Науково-дослідного інституту інтелектуальних комп'ютерних систем Тернопільського національного економічного університету (ТНЕУ) к.т.н., професора кафедри ІОСУ Кочана В.В., завідувача кафедри інформаційно-обчислювальних систем і управління, д.т.н., професора Саченка А.О. та директора Науково-дослідного інституту інноваційного розвитку та державотворення ТНЕУ Мартинюка В.Ф., створена для приймання роботи, виконаної в рамках тематичного плану науково-дослідних робіт ТНЕУ на тему «Виділення контурів для структурно-статистичної ідентифікації об'єктів» (номер державної реєстрації – 0110U000786), встановила:

1. Розроблена Загородньою Д.І. система виділення контурів для структурно-статистичної ідентифікації об'єктів базується на запропонованих у її дисертаційній роботі методі контурної сегментації на базі методу Канні та методі ієрархічної структурно-статистичної ідентифікації об'єктів.
2. Проведені за участю Загородньої Д.І. експериментальні дослідження дали змогу здійснити перевірку запропонованої інформаційної технології розпізнавання об'єктів системою відеоспостереження.
3. Експериментальні дослідження показали, що розроблена інформаційна технологія дозволяє зменшити обсяг оброблюваної інформації під час процесу виявлення об'єктів системою відеоспостереження.

Директор НДІ інтелектуальних
комп'ютерних систем,
к.т.н., професор кафедри ІОСУ — Кочан В.В.

Завідувача кафедри інформаційно-
обчислювальних систем і управління,
д.т.н., професор — Саченко А.О.

Директор НДІ
інноваційного розвитку та державотворення — Мартинюк В.Ф.