

ВІДТВОРЕННЯ ТЕПЛОВОГО ЗОБРАЖЕННЯ НА МОНІТОРІ ПЕРСОНАЛЬНОГО КОМП'ЮТЕРА

© Боженко В.І., 2009

Пропонується комплекс математичного і програмного забезпечення для збирання, збереження й опрацювання графічних даних, отриманих від тепловізійної камери.

The complex of the mathematical support and software for collection, storage and processing of the graphic data, received from the thermovision camera, is offered.

Вступ. Бурхливий розвиток комп'ютерних технологій дав потужний поштовх до розвитку суміжних галузей, зокрема, телевізійних технологій. Кожне нове досягнення у одній із цих технологій надає нові можливості в іншій і водночас висуває до неї нові вимоги. Відтак актуальною стає задача включення тепловізійної камери (ТПК) до складу комп'ютеризованої системи з метою підвищення ефективності засобів дистанційного моніторингу [1].

Аналіз досліджень у цій галузі показав, що сьогодні не тільки не існує стандартного програмного забезпечення (ПЗ) для ТПК, але й серед фахівців цієї галузі не існує усталеного погляду на те, якою саме повинна бути оптимальна форма представлення отриманої інформації, що задовольняла б усі вимоги споживача та відповідала спроможності виробника [2]. До того ж практично кожна модель ТПК формує масив графічних даних для перегляду та зберігання у власному, специфічному форматі, який не підтримує ПЗ жодного іншого виробника [3, 4].

Метою даної статті є аналіз шляхів покращання якості теплового зображення на екрані монітора персонального комп'ютера (ПК) за допомогою засобів математичного та програмного забезпечення для збирання, відтворення та зберігання у загальноприйнятному графічному форматі інформації, отриманої від ТПК.

1. Формат початкового теплового зображення

У цьому випадку розглянута ТПК «Піровідикон-3», що формує цифрові дані у специфічному форматі, який має розмір 32040 байтів та описує зображення розмірністю 178 x 180 пікселів. Кожен піксел кодується однією з 64 градацій яскравості формованого зображення. З огляду на можливості найживаніших сьогодні вітчизняних ТПК та на найпоширеніші вимоги до теплового моніторингу у різних галузях діяльності людини, цей розклад приймають близьким до оптимального [5].

Під час сумісної роботи з ПК масив даних, сформований у ТПК, передається у ПК, де формується у файл. Цей файл має розширення *.bin, а його ім'я – порядковий номер кадру ТПК. Значення пікселів зображення зберігаються неперервним рядком за їх розташуванням у зображенні зліва направо, починаючи з верхнього рядка.

Оскільки зазначені дані є нормованими для кожного файла цього специфічного формату, що визначається за двома ознаками – розширенням та розміром файла, то ця інформація не вказується у заголовку файла, як це звичайно прийнято у графічних форматах даних, а повинна бути передбачена у програмі обробки цих даних.

2. Генерування початкового зображення

На початку роботи в оперативній пам'яті ПК відводиться місце для масиву розмірністю 178 x 180 байтів, в який буде зчитано початкове теплове зображення. Після цього відбувається

пошук файла на диску та перевірка його відповідності заданим параметрам – розширенню та розміру. Якщо файл їм відповідає, його вміст переписується до відведеного масиву.

Необхідно зауважити, що опрацювання зображень, отриманих з інфрачервоного спектра, має свою специфіку, оскільки воно пов'язане не з дослідженням візуального контрасту, а просторовим розподілом і зміною в часі енергетичної (теплової) яскравості досліджуваних об'єктів. При цьому необхідно задовольнити дві доволі взаємосуперечливі вимоги: з одного боку, необхідно виявити зони з аномальним тепловим розподілом, а з іншого – ідентифікувати об'єкт загалом.

Одним з основних засобів задоволення першої вимоги є доповнення теплового зображення градаційною шкалою – т.зв. “тепловим клином”. Тому при роботі з тепловим зображенням в оперативній пам'яті ПК відводиться місце для масиву розмірністю 188 x 180 байтів: основний масив призначений для перезапису у нього побітових значень з першого (вихідного) масиву, а у додаткові 10 рядків заноситься градаційна шкала яскравості з діапазоном значень від чорного (0/0/0) до білого (63/63/63).

Блок, відведений для градаційної шкали розмірністю 180 x 10 пікселів, розбивається на 20 частин розміром 4 x 10 пікселів і, починаючи з верхнього лівого кута, кожна наступна частина заповнюється значенням, що на одиницю більше за значення вмісту попереднього блоку. Оскільки зміна градації відбувається через кожні 4 пікселі, то при зоровому сприйнятті градаційна шкала являє собою плавну зміну відтінку – від чорного до білого, не включаючи у себе граничні значення яскравості.

3. Корекція контрастності

Коли вирішене питання представлення стандартної градації базового кольору, виникає інша проблема, яка зазвичай передбачена у ПЗ скануючого обладнання, а саме можливість зручної для оператора ручної корекції яскравості та контрастності зображення. При цьому, оскільки їх основний підбір та встановлення проводяться безпосередньо при налагодженні ТПК, від ПЗ вимагається забезпечити лише можливість додаткової корекції контрастності оцифрованого зображення.

Для цього вводимо початкове значення контрасту (у даному випадку 100) і передбачаємо можливість зміни цієї величини у більший чи менший бік на 30%, оскільки при більшій зміні контрастності, яка попередньо вже корегувалася апаратно, виникає загроза зробити зображення візуально несприйнятним.

Це значення у випадку, коли воно буде змінюватись оператором у процесі роботи, заноситься до файла установок програми, звідки воно зчитується на початку роботи – для того, щоб не виникала потреба щоразу вручну підбирати вже знайдені параметри. Виведення значення контрастності за встановлені у програмі межі можливе у разі введення цього числа до файла програмних установок за допомогою текстового редактора.

Після зчитування з файла ініціалізації значення зміни контрастності кожен піксел зображення коригується відповідно до нього за формулою

$$A [y, x] := \lfloor A [y, x] \cdot 100/B \rfloor, \quad (1)$$

де $A [y, x]$ – значення яскравості зображення у точці з координатами $[y, x]$; B – значення зміни контрастності зображення.

За адресою кожного нового значення пікселя заноситься ціла частина від результату ділення.

4. Створення негативного варіанта зображення

У результаті експериментального аналізу теплових зображень встановлено, що їх негативне представлення утворює свого роду “імітацію” зображення, отриманого у видимому діапазоні, і тому порівняно з вихідним ефективніше піддається ідентифікації.

Створення негатива зображення полягає в інверсії значення яскравості кожного його пікселя на протилежне. Оскільки значення контрастності зображення знаходиться у межах від чорного

(значення 0) до білого (значення 63), то інверсія значення для кожної точки відбувається за формулою

$$A [y, x] := 63 - A [y, x], \quad (2)$$

де $A [y, x]$ – значення кольору зображення у точці з координатами $[y, x]$.

Оскільки градаційна шкала яскравості була попередньо введена до складу зображення, інверсія так само стосується і її.

5. Псевдокольорове розфарбування зображення

При виявленні зон з аномальним тепловим розподілом одним з найпотужніших засобів вважають т.зв. псевдокольорове розфарбування (ПКР). Це означення має на увазі відсутність наперед заданого жорсткого закріплення розфарбування перетворюваного зображення відносно реального. ПКР створює таку кольорову карту досліджуваного об'єкта, яка унаочнює якісь його певні властивості (наприклад, теплову сигнатуру) без жодного зв'язку з реальним кольором його поверхні (у будь-якому діапазоні) та має на меті привернути увагу оператора до областей зображення, що мають, наприклад, певний, слабо відмінний від суміжних областей, тепловий розподіл.

Тут необхідно зауважити, що звичайно людське око може сприймати до 17 тис. кольорів, а з урахуванням яскравості – до 10 млн. Але якщо йдеться не про сприйняття, а про ідентифікацію, кількість цих градацій різко зменшується. При, наприклад, монохромному відтворенні оператор здатен ідентифікувати не більше 8 напівтонів. Саме тому із застосуванням ПКР інформативність зображення різко підвищується, зокрема, уможливується розрізнення у суміжних температурних зонах. Але, з іншого боку, при надмірному рівні ПКР зображення набуває складної кольорової структури (особливо – при порушенні принципу монохромності, яскравості та безперервності у 3-вимірному кольоровому просторі), однозначність інтерпретації якої порушується.

Тому найефективнішою вважають кількість градацій кольорів для розфарбовування не більшу 64 за умови компенсації фонових спотворень до 3...5% [6]. Крім того, для зменшення впливу спотворюючого ефекту ПКР бажано:

- доповнювати зображення кольоровим клином;
- забезпечувати можливість вибору певного обмеженого діапазону ПКР з загальної кількості рівнів квантування, тобто робити кольорові “зрізи” – наприклад, при загальному ПКР у 256 кольорів додатково робити вибірку з 16-ти;
- розширювати палітру не за рахунок кількості кольорів, а їхньої яскравості;
- забезпечення можливості перебудови алгоритму ПКР з метою вибору оптимального для кожного певного зображення.

З огляду на зазначене у пам'яті ПК формується додатковий масив – палітра кольорів, в якому адресою кожної комірки є значення відтінку сірого кольору, а її вмістом – значення кольорового пікселя, що має 3 складові – значення кольорів RGB-схеми у цій точці. Ці дані доцільно зберігати в окремому файлі та читувати їх на початку роботи.

При створенні нового масиву даних в оперативній пам'яті ПК на місце кожного пікселя зображення заноситься не значення відповідного пікселя старого, чорно-білого зображення, а ті дані, що знаходяться у “палітрі” за адресами, які і є значеннями відтінків сірого кольору.

6. Збереження отриманих зображень

Для збереження і подальшого опрацювання всіх зображень, отриманих у процесі роботи програмного комплексу, необхідно їх транслювати до одного з загальноприйнятих графічних форматів представлення растрових даних.

З великої кількості поширених сьогодні форматів було обрано формат BMP як найбільш “природний” у середовищах з операційною системою (ОС) MS Windows. Він використовується, коли вимагається якнайшвидше відобразити графічну інформацію: оскільки об'єкт переміщується безпосередньо з оперативної пам'яті, то у цьому форматі він може бути відображений швидше, ніж при розпізнаванні кодової послідовності символів з подальшим відтворенням потрібного зображення.

Для запису у форматі BMP потрібні:

- масив значень кольорів кожного пікселя зображення;
- ширина та висота зображення у пікселях;
- кількість кольорових площин;
- кількість розрядів графічного файлу.

Всі ці дані не потребують спеціального обчислення, оскільки є такими, що залежать від опцій, встановлених у ТПК для оцифрування зображення, отриманого у процесі її роботи (див. п.2), а відтак заздалегідь обумовленими під час розроблення ПЗ. Потім за допомогою засобів програмування MS Windows можливо створити графічні BMP-файли, що відповідатимуть цим параметрам.

7. Вибір мови програмування

Попри те, що останнім часом надзвичайно поширеною стала ОС MS Windows, деякі обчислювальні комплекси продовжують працювати за останніми версіями MS DOS. Тому досі доцільним є створення двох варіантів програмного забезпечення – під DOS та під Windows. При цьому кожен варіант розробки буде мати як недоліки, так і переваги. Звичайно, програмна система, розроблена для середовища MS Windows, забезпечує зручний користувацький інтерфейс і можливість в процесі роботи користуватися усіма ресурсами та можливостями, що пропонує Windows. А програма, розрахована на роботу у середовищі MS-DOS, забезпечує значно вищу швидкодію (при рівноцінних апаратних засобах) та займає значно менший обсяг оперативної пам'яті, що у деяких випадках є життєво важливим.

Оскільки основна робота ПЗ пов'язана з великими масивами інформації та вказівниками на них, для розроблення програмного комплексу у середовищі MS-DOS доцільно використовувати мови C++, компілятор якої забезпечує швидку програмну обробку великої кількості циклів та багатовимірних масивів.

Для розроблення ж ПЗ для роботи у середовищі MS Windows обрано мову Delphi, яка має у своїй основі структуру, близьку до мов Pascal та C++, що дає змогу використати розробки, створені для DOS-варіанта програми. Водночас Delphi дає змогу під час програмування користуватися ресурсами Windows для створення графічного користувацького інтерфейсу, візуального програмування за допомогою механізму “drag-and-drop”, вводити до складу розроблюваної програми компоненти з бібліотеки взірців та під'єднувати готові програми до складу ПЗ MS Office.

Отже, комплекс ПЗ для обробки зображення, отриманого від ТПК, складається з блоків програм, написаних мовами C++ та Delphi, та файлу текстового формату *rab.pal*, в якому зберігаються кодові значення 64 кольорів схеми RGB.

8. Алгоритм цифрової обробки теплових зображень

Відповідно до описаних вище процедур пропонується оригінальний алгоритм цифрової обробки теплових зображень (рис. 1), згідно з яким відбувається:

1. Зчитування з файлів *rab.pal* та *dh.ini* значень контрастності та параметрів кольорової RGB-схеми зображення.
2. Зчитування початкового теплового зображення з файла **.bin*.
3. Корекція контрасту теплового зображення.
4. Доповнення зображення градаційною шкалою.
5. Створення негативного варіанта зображення.
6. Створення псевдокольорового варіанта зображення.
7. Виведення трьох отриманих зображень на монітор ПК.
- 8–9. За необхідності подільшої корекції – зміна оператором цих значень та їх збереження у файлі установок програми *dh.ini*.
10. Аналіз наявних на диску імен теплових зображень, обчислення та генерування імен для новостворених графічних файлів.
11. Створення графічних файлів у форматі BMP.
- 12–14. Запис на жорсткий диск ПК варіантів теплового зображення.

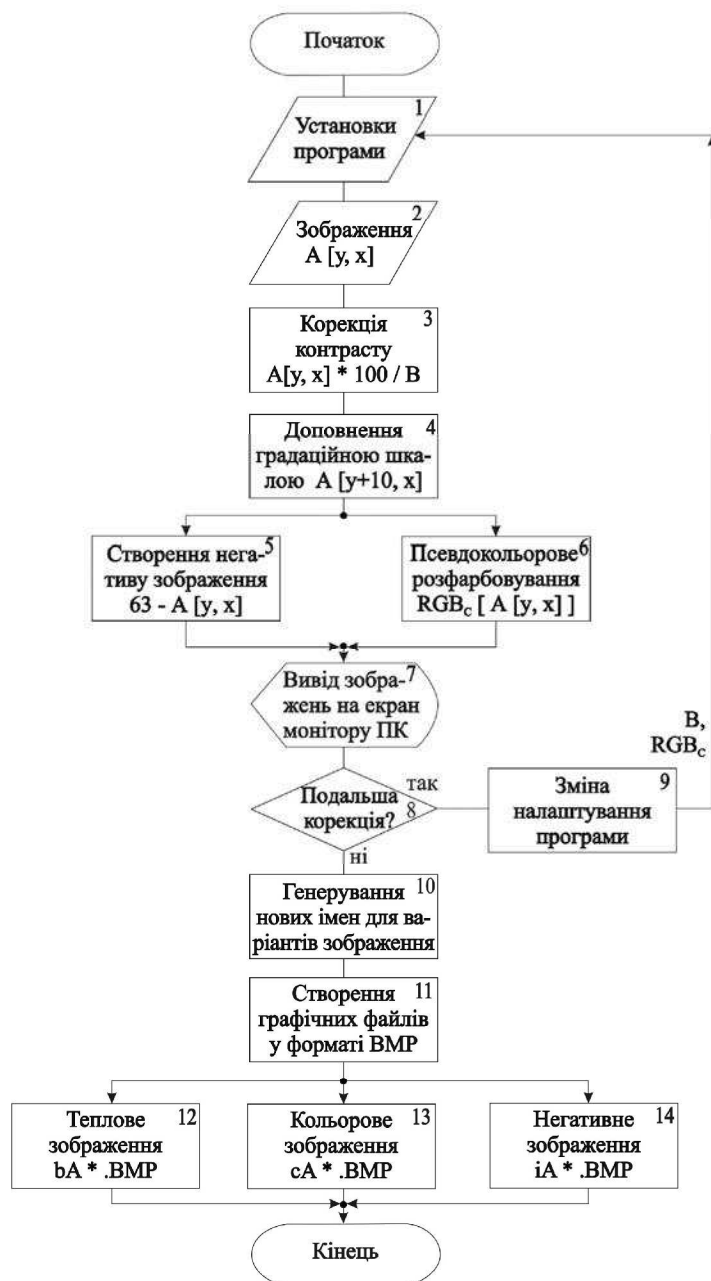


Рис. 1. Граф-схема алгоритму цифрової обробки теплових зображень

Отже, на основі початкового теплового зображення для аналізу та прийняття рішень оператор отримує три графічні зображення: скореговане за контрастністю та доповнене градаційною шкалою теплове зображення, а також його негативну та псевдокольорову версії.

9. Програмна реалізація алгоритму для MS-DOS

На початку роботи програми доцільно відвести у оперативній пам'яті місця для масивів кольорів $RGB_{pal}[256, 3]$ та вхідних даних $hp[178, 180]$.

Першим кроком роботи програми є перевірка наявності вхідних даних при запуску у командному рядку – змінна $argc$. Потім проводиться ініціалізація графічного режиму VGA.

Функція $init_vga_pal()$ виконує завантаження у пам'ять значень кольорів моделі RGB, що містяться у текстовому файлі $rab.pal$ (див. п.5).

Якщо при запуску програми у командному рядку не було задане ім'я файла для опрацювання ($funk=FALSE$), то відбувається пошук у поточному каталозі першого з файлів, розмір якого є 32040

байти, а розширення – *bin*. З нього зчитується масив вхідних даних за адресою *hp [0..177, 0..179]* відповідно до п.2.

Управління передається до функції *keyboard ()*, яка чекає на перше натискання клавіші, що заноситься у змінну *key*. Подальші дії відбуваються при натисканні оператором ключових клавіш <1>, <2>, <3>, <S>, <Esc>.

При *key='1'* відбувається побітове виведення на екран з пам'яті зображення, що міститься у масиві даних *hp []*.

При *key='2'* екран очищується, і на нього виводиться вміст масиву *hp []*. Попередньо значення кожної комірки масиву обраховуються відповідно до (1).

При *key='3'* екран очищується, і на нього виводиться обрахований вміст масиву. У масиві *pal [0..255, 0..2]* зберігаються значення кольорів, що були занесені туди з текстового файлу у результаті виконання функції *init_vga_pal ()*.

При *key='S'* відбувається запуск функції *bin2bmp ()*. До масиву *head* заносяться значення інформаційного заголовка графічного файлу. Створюється (відкривається для запису) файл *fnb* (цій змінній присвоюється значення імені файлу, з якого дані були зчитані) і у нього записується масив *head*. Після цього до файла поступово (за допомогою циклу) записуються дані про графічний об'єкт з оперативної пам'яті. На цьому створення графічного файлу завершено.

У випадку, коли ім'я файла для обробки було задане оператором у командному рядку, на цьому робота програми завершується. Якщо ж пошук файла було здійснено програмно, керування передається до функції *receive_file*. Шукають та підраховують усі файли у поточній директорії, що відповідають заданим параметрам. Результат підрахунку заноситься у змінну *cnt*. Оскільки в іменах файлів, що передаються до ПК від ТВК, імена (крім першої літери) є їх порядковими номерами, одночасно до змінної *vc* “вирізаються” цифри з імені файла, перетворюються до числового (*int*) формату і порівнюються з лічильником кількості потрібних файлів. Якщо ці два значення збігаються, обрахована кількість вважається правильною. У протилежному випадку лічильник кількості файлів скидається у “1” (мається на увазі файл, що вже був опрацьований).

Після цього циклу у випадку, коли *cnt>1*, запускається цикл від *i := 2* до *cnt* (за кількістю знайдених файлів), у якому на кожній його стадії повторюються всі вищезазначені дії. Завершується робота програми, коли всі наявні файли кількістю *cnt* буде оброблено або оператор натисне клавішу <Esc>.

Загальний текст DOS-варіанта ПЗ складатися з 359 рядків.

10. Програмна реалізація алгоритму для MS Windows

Відповідно до правил створення ПЗ для середовища MS Windows, цей програмний комплекс складається з головної екранної форми (рис. 2), з завантаження якої починається робота системи, та ряду процедур та функцій, що використовуються в процесі роботи залежно від дій оператора. Екранна форма програми має:

- 3 вікна для виведення результуючих зображень;
- заголовок програми, у якому вказуються імена початкового та результуючих файлів;
- список розширень вхідних файлів (складається з одного елемента – “*.bin”);
- програмну компоненту, призначену для пересування курсора по дереву каталогів для вибору файлів;
- список досяжних логічних дисків;
- різновид лінійки прокрутки для вибору коефіцієнта зміни контрастності зображень.

Всі ці елементи побудовані за допомогою загальносистемних ресурсів MS Windows з використанням ресурсів Бібліотеки Компонент, наданої авторами мови Delphi.

Як кожна ужиткова Windows-програма, ПЗ має файл ініціалізації *dh.ini*, з якого на початку роботи завантажуються початкові установки програми – адреса останньої роботи з файлами *fName*, значення зміни контрастності (див. п.2) *iMx* та номер COM-порта *port* для введення інформації від ТПК.

Першим кроком роботи програмного комплексу є виведення на екран вікна діалогу (рис. 3).

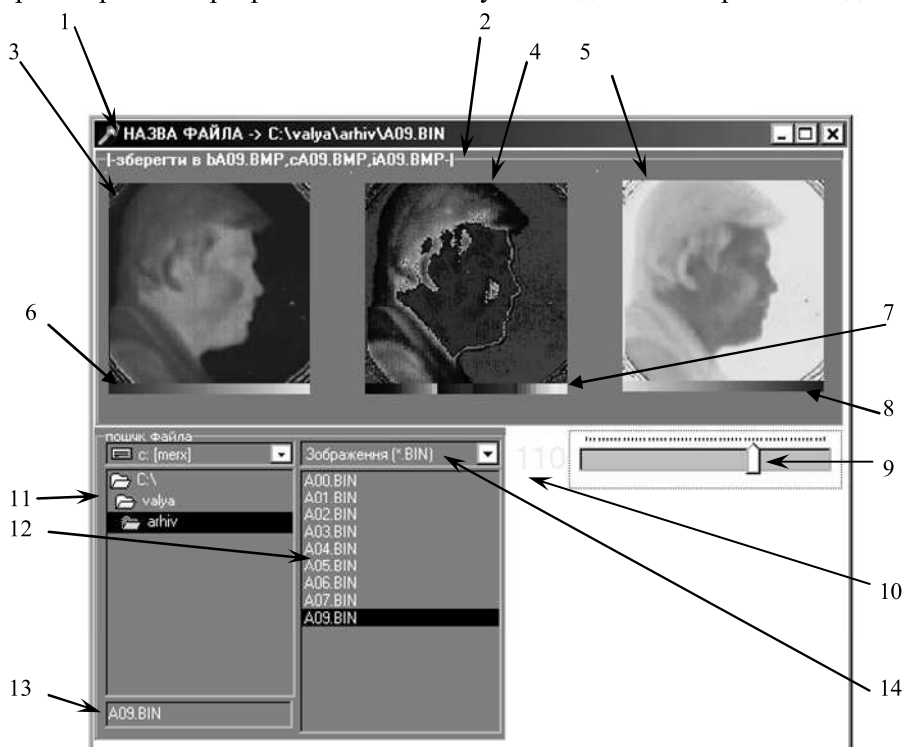


Рис. 2. Робоче вікно програмного комплексу. Елементи вікна:

- 1 – заголовок вікна програми з іменем оброблюваного файла;
- 2 – список імен результуючих файлів, на які перетворюється початкове зображення;
- 3 – початкове зображення; 4, 5 – псевдокольоровий та негативний варіанти зображення;
- 6, 7, 8 – градаційні шкали; 9 – лінійка прокрутки для зміни значення контрастності;
- 10 – поточне значення контрасту; 11 – дерево каталогів поточного диску;
- 12 – список файлів у поточному каталозі;
- 13 – назва файла, що опрацьовується; 14 – активне розширення вхідних файлів

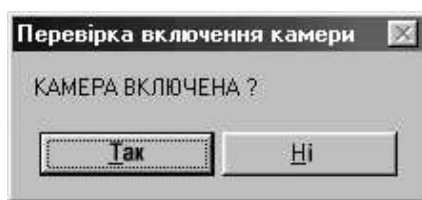


Рис. 3. Початковий запит про режим роботи програмного комплексу

У випадку, коли робота ПЗ має відбуватися синхронно з роботою ТПК (оператор натискає кнопку <Так>), виконується запуск функції *bioscom*, що є вставкою мовою Асемблер, яка програмує порт СОМ-2 для зчитування інформації від ТПК. За допомогою функції *Inportb* та процедури *Outportb* оцифрована інформація передається у вигляді описаного у п.7 вхідного масиву даних від ТПК на монітор ПК.

У випадку, коли ПЗ працює з кадрами, вже переданими та збереженими у ПК (оператор натискає <Ні>), виводиться на екран головна екранна форма та запускаються процедури, що до неї підключені.

Коли у вікні дерева каталогів вибрано файл з розширенням *.bin, він відкривається для читання, і дані з нього переписуються у масив *bvv* [0..177, 0..179], після чого копіюються у масив

hp [178, 180]. Назва файлу зі змінної *fName* виводиться у заголовок вікна програми. Ці дії виконує процедура *Create_BMP*.

У масив *bt* [0..187, 0..179] відповідно до пп. 2, 3 переписується масив вихідних даних та дописується градаційна шкала, яка відповідає температурі досліджуваного об'єкта, починаючи від мінімального та закінчуючи максимальним значенням температури. За (1) проводиться корекція контрастності, значення якої у змінну *iMx* було зчитане процедурою *ReadIniFile* при запуску програми з файлу *dh.ini*.

Вміст масиву *bt* копіюється у масиви:

- *wB* [188, 180], причому значення кольору кожного пікселя збільшується на 16 з метою підвищення чіткості зображення для покращання його зорового сприйняття;
- *wI* [188, 180] копіюється вміст масиву *bt*, причому значення кольору кожного пікселя обраховується за (2) для отримання його негативної копії;
- *wC* [188, 180], причому попередньо виконується процедура *FormPalette* – з файлу *rab.pal* у масив *aRGB* [0..255, 0..2] завантажуються значення кольорів схеми RGB і за його допомогою у *wC* записуються значення пікселів, які відповідають сформованому кольоровому зображенню.

За допомогою стандартної Windows-функції *CreateBitmap* значення цих трьох масивів перетворюються на BMP-зображення, що виводяться у відповідне місце екранної форми програми та одночасно зберігаються у вигляді файлів.

За допомогою лінійки прокрутки оператор у процесі роботи має змогу змінювати значення змінної *iMx*, а також одночасно з цим перераховуються та оновлюються на екрані і у файлах усі три масиви – *wB*, *wC*, *wI*.

За допомогою процедури *Paint_Lbl_on_gpPict* у робочому полі програми генеруються імена файлів, в які записуються результуючі зображення. Останні три літери імені файлу (його розширення) замінюються на "bmp", а перед іменем додаються літери "b", "c" або "i" для чорно-білого, кольорового або інверсного зображення відповідно.

Отже, в процесі підбору контрастності зображення на екрані та на диску створюються кожен раз нові варіанти зображення. Коли процедура *Receive_file* викликає процедуру *Create_BMP* з іменем нового файлу з розширенням *.bin як вхідний параметр, ПЗ розпочинає дії з обробки нового зображення, і процес повторюється знову.

Робота програми завершується за стандартною для Windows-ужитків схемою – натисканням комбінації клавіш <Alt + F4> або кнопкою закриття ужитку. Керування передається до стандартної процедури *FormDestroy*. Вона викликає процедуру *WriteIniFile*. До файлу ініціалізації *dh.ini* записуються ті значення, що були зчитані на початку роботи програми та могли зазнати змін. Область пам'яті *ini.INF* звільняється. Оскільки BMP-файли вже були створені в процесі роботи, на цьому виконання програмного комплексу завершується, а усі області пам'яті, що були зайняті значеннями її змінних, звільнюються.

Сукупний текст усіх процедур та функцій ПЗ становить 466 рядків.

Висновки. ПЗ комп'ютеризованого комплексу обробки теплових зображень залежно від вибору оператора, що керує ним за допомогою ПК, виконує такі дії:

1. Приймання та запам'ятовування у файли даних, отриманих з ТПК.
2. Пошук файлів – результатів роботи ТПК за їх іменем та розміром і їх виведення на екран.
3. Корекцію контрастності отриманих зображень.
4. Створення негативного та псевдокольорового еквіваленту отриманих зображень.
5. Збереження створеної графічної інформації шляхом створення нових файлів у форматі BMP.

Напрямами подальшого вдосконалення ПЗ може бути:

1. Розширення функціональних можливостей алгоритму псевдокольорове розфарбування.
2. Підвищення роздільної здатності формованого теплового зображення, яке уможливується застосуванням новітніх надчутливих піровідиконів.
3. Доповнення теплового зображення гістограмами, зрізами тощо.

4. Перетворення теплової сигнатури на температурну.
5. Формування комплексного зображення.

1. Боженко В.І., Бродський М.О., Кондратов П.О., Шклярський В.І. Особливості дистанційного моніторингу за допомогою тепловізійної камери, обладнаної каналом видимого зображення // *Радіоелектроніка та телекомунікації*. – 2007. – № 595. – С. 28–32. 2. Hrytskiv Z., Kondratov P. The ways of multispectral devices construction for thermal object visualization // *Telecommunications in Modern Satellite, Cable and Broadcasting Services. Proceedings*. – Nis: Galeb, 1999. – Vol.1. P.184, 185. 3. Colbert F. Understanding Proprietary Infrared Image Files [Electronic resource] // *Accolade Group Internet Publishing, 2009*. – Mode of access: www.irinfo.org/Articles/article_4_2006_colbert.html. – Last access: 2009. – Title from the screen. 4. Саликов В. Л. Эпоха ночной войны. Обзор международного рынка приборов ночного видения // *Специальная техника*. – 2000. – № 5. – С. 21–32. 5. Колобродов В., Шустер Н. Тепловізійні системи. – К.: НТУУ "КПІ", 2001. – 340 с. 6. Мазуров А., Николаев Е. Искажения при цветовом кодировании черно-белых изображений // *Техника средств связи. Сер. ТТ*. – 1984. – №1. – С.59–60.

УДК 681.3, 621.3

О.Ю. Бочкарьов, В.А. Голембо

Національний університет "Львівська політехніка",
кафедра електронних обчислювальних машин

СПОСОБИ ОРГАНІЗАЦІЇ ПЕРЕМІЩЕННЯ МОБІЛЬНИХ ВИМІРЮВАЛЬНИХ АГЕНТІВ: ПІДХОДИ ДО ПОБУДОВИ КОНЦЕПТУАЛЬНИХ МОДЕЛЕЙ

© Бочкарьов О.Ю., Голембо В.А., 2009

Розглянуто проблему побудови концептуальних моделей, за допомогою яких можна отримувати універсальні з погляду специфіки способу організації переміщення вимірювального агента розв'язання важливих задач автономних досліджень.

The problem of desing of conceptual models allowing to find general solutions of important autonomous explorations' tasks with diverse explorer agent's mobility technics is considered.

Вступ. Останнім часом посилюється інтерес до автономних інтелектуальних систем, здатних розв'язувати складні задачі самостійно без участі людини [1,2], зокрема до розподілених систем автономних досліджень [3–5]. Різке зростання доступної обчислювальної потужності та комунікаційних можливостей для побудови таких систем усунуло цілу низку проблем, які раніше здавалися непоборними. Однак водночас залишаються невирішеними або далекими від остаточного вирішення декілька ключових проблем організації роботи відповідних систем. Однією з таких проблем є проблема організації переміщення автономних дослідницьких станцій (вимірювальних агентів [3–5]) у складі розподіленої системи автономних досліджень. Відтак в роботі розглядається актуальне питання розв'язання цієї проблеми шляхом побудови концептуальних моделей на основі узагальненої схеми організації переміщення та операційної схеми переміщення мобільного вимірювального агента.

Стан проблеми. Здатність вимірювального агента переміщуватись у просторі відкриває широкі можливості щодо глибшого вивчення об'єкта дослідження за рахунок адаптивного управління послідовністю розміщень або траєкторією переміщення як окремого вимірювального агента, так і усього колективу вимірювальних агентів [5–9]. Однак переважно дослідники та