

Висновок

Новий метод розміщення вершин деревоподібних графів, заданих матрицями суміжності, дає змогу ефективно розташувати вершини графу як на екрані монітора, так і на шпальті видання з забезпеченням сформованих критеріїв візуалізації. Отримані за його допомогою зображення графів на площині мають високу наочність, що дає змогу адекватно оцінювати представлену ним структуру.

1. Басюк Т.М. Аналіз та класифікація методів візуалізації // Поліграфія і видавнича справа. – 2003. – Вип. 40. – С. 109–114. 2. Басюк Т.М. Критерії відображення графів в процесі візуалізації // Наукові записки УАД. – 2004. – Вип. 7. – С. 60–63. 3. Басюк Т.М. Метод розміщення вершин графа в процесі візуалізації // Вісн. Нац. ун-ту “Львівська політехніка”. – 2004. – № 519. – С. 3–10. 4. ДСТУ 3018-95. Видання. Поліграфічне виконання. Терміни та визначення. – К.: Держстандарт України, 1995. – 30 с. 5. ДСТУ 3772-98. Оригінали для поліграфічного відтворення. Загальні технічні вимоги. – К.: Держстандарт України, 1998. – 54 с. 6. Дунець Р.Б. Аналіз та синтез топологій комп’ютерних видавничо-поліграфічних систем: НВФ „Українські технології”. – Львів, 2003. – 192 с. 7. Касьянов В.Н., Евстегнеев В.А. Графы в программировании: обработка, визуализация и применение. – СПб.: БХВ, 2003. – 1104 с. 8. Харари Ф. Теория графов. – 2-е изд. – М.: УРСС, 2003. – 296 с. 9. Хорн Р., Джонсон Ч. Матричный анализ. – М.: Мир, 1999. – 655 с.

УДК 004.932

О.О. Карпін

Національний університет “Львівська політехніка”,
кафедра електронних обчислювальних машин

РЕКОНСТРУКЦІЯ ПОВЕРХОНЬ НА РАСТРОВИХ ЗОБРАЖЕННЯХ ТРИВИМІРНИХ ОБ’ЄКТІВ

© Карпін О.О., 2006

Запропоновано алгоритм розв’язування проблеми автоматичної реконструкції поверхонь на растрових зображеннях тривимірних об’єктів. Показано ефективність запропонованого алгоритму на тестових прикладах.

Algorithm of the automatic three-dimensional objects reconstruction from the raster images is proposed. The test images show the effectiveness of the proposed algorithm.

Вступ

Під час розпланування та розбудови населених пунктів, для розв’язання задач телекомунікацій, картографування, у лісовій та гірській справах тощо широко використовують растрові зображення, отримані засобами активної оптичної та лазерної альтиметрії [1–7]. Сучасний розвиток неконтактних засобів вимірювання дальності дає змогу отримати ці зображення з великою точністю та роздільною здатністю. До того ж дані, як правило, скануються з великої ділянки земної поверхні, площа якої може сягати декількох десятків квадратних кілометрів. Для виконання тривимірної візуалізації таких великих растрових зображень тривимірних об’єктів, розмір яких часто перевищує обсяг оперативної пам’яті комп’ютера, необхідно використовувати спеціалізовані графічні станції та затратити значний час на виконання цієї операції. При цьому операція маніпулювання тривимірними об’єктами потребує точної ручної роботи оператора щодо виділення контурів об’єкта та їхньої сегментації. З метою автоматизації цих процесів необхідно аналітично описати зображення кожного тривимірного об’єкта та створити тривимірну модель сцени.

Аналіз останніх досліджень та публікацій

Протягом останніх років для вирішення поставленої проблеми було розроблено низку методів. Однак вони, як правило, потребують додаткових апріорних відомостей про сам об'єкт реконструкції, зокрема апріорної інформації про топологію досліджуваної поверхні, інформації про її орієнтацію у просторі, вимагають карти місцевості або втручання оператора в процес реконструкції [1, 2]. Крім того, більшість запропонованих методів реконструкції поверхонь описують форми об'єктів за допомогою базових примітивів або працюють тільки з поверхнями першого порядку [2–6]. Ці методи добре працюють при реконструкції форми будинків та погано пристосовані для роботи з поверхнями складної форми. Останнім часом швидко зростає точність та роздільна здатність неконтактних засобів вимірювання дальності з повітря. Так, за допомогою гелікоптера у 1997 році можна було отримати зображення квадратного метра площі з точністю 0.3 м за планіметрією та 0.1 м за висотою [8]. Сьогодні можна отримати висотні дані для 278 точок на квадратний метр з точністю декілька сантиметрів як за планіметрією, так і за висотою [7]. Тому виникає потреба врахування цих додаткових даних для точнішого відновлення форми досліджуваної поверхні, ніж це можуть забезпечити методи на базі примітивів.

Постановка задачі

У статті запропоновано алгоритм вирішення проблеми автоматичної реконструкції поверхонь на растрових зображеннях тривимірних об'єктів, який враховує сучасний стан неконтактної вимірювальної техніки, дає змогу описувати поверхні довільної форми та не потребує додаткових апріорних відомостей про об'єкт і втручання оператора в процес реконструкції. Наведені тестові приклади доводять ефективність запропонованого алгоритму.

1. Алгоритм автоматичної реконструкції поверхонь на растрових зображеннях тривимірних об'єктів (РЗТО)

Алгоритм розв'язування поставленої задачі розділено на чотири фази [8]:

1. Виділення країв локальних об'єктів (ЛО) на РЗТО;
2. Відокремлення та опис поверхні кожного ЛО щільною трикутною сіткою (ТС);
3. Спрощення подання поверхні кожного ЛО до заданого рівня деталізації;
4. Побудова тривимірної моделі території.

Ця схема дасть змогу використати переваги ТС для подання поверхонь довільної форми, згладити плоскі ділянки поверхонь ЛО та надасть задачі реконструкції поверхонь повної автоматичності у виконанні.

Фаза алгоритму виділення країв ЛО на РЗТО

Як вхідні дані етап виділення країв ЛО використовує РЗТО та мінімальні параметри об'єктів реконструкції (висота, площа, градієнт та інші). Оскільки задача виділення країв передбачає диференціювання даних, що містять шум, то з метою обчислення похідних використовуємо операцію згортки диференціального оператора Лапласа із вихідним РЗТО, з подальшим застосуванням операції фільтрування гауссівською функцією [9,10]. При цьому для пришвидшення обчислень можна застосувати ядро лапласіана гауссіана $\nabla^2 G_\sigma$ та за заданими мінімальними параметрами об'єктів реконструкції вибрати необхідний масштабний рівень σ . Весь процес автоматичного виділення контурів локальних об'єктів на РЗТО разом з процедурами потоншення контурів та замикання відкритих контурів детально розглянуто в [9].

Алгоритм опису поверхні ЛО щільною регулярною ТС на РЗТО та растровому зображенні контурів тривимірних об'єктів (РЗКТО)

Існує велика кількість методів розв'язання задачі триангуляції [11–13]. Для побудови щільної регулярної ТС найкраще підходять коміркові методи триангуляції (cell-based), а в цьому випадку

для 2,5-вимірному простору – метод маркованих квадратів (marching square) [12,13]. Цей клас методів є швидким (асимптотична складність $O(N)$ операцій) та простим у використанні, проте він швидше розрахований на виділення ізоповерхні в просторі, ніж для побудови РТС конкретного тривимірного об'єкта. До того ж процедуру відокремлення ЛО взагалі не описано для цього класу методів. Застосування іншого класу методів триангуляції, наприклад, на базі триангуляції Делоне (асимптотична складність $O(N^2)$, рідше $O(N \log N)$ операцій), значно ускладнить процес обробки [11]. В роботі запропоновано процедуру відокремлення ЛО та алгоритм опису поверхні ЛО щільною регулярною ТС на РЗТО та РЗКТО на основі методу триангуляції маркованих квадратів.

Як вхідні дані алгоритм використовує два растрові зображення однакового розміру: РТЗО та РЗКТО. Причому контури у РЗКТО попередньо мають бути:

- виділені одним кольором (білі контури на чорному фоні);
- замкнені;
- потоншені до товщини контуру в один піксель.

Оскільки алгоритм для своєї роботи використовує регулярність даних растрового зображення та продукує регулярну ТС, то для представлення вихідної триангуляції достатньо найпростішої структури:

- список вершин;
- список трикутників.

Надлишковість такого представлення є мінімальною. Запропонований алгоритм опису поверхні ЛО щільною регулярною ТС на РЗТО та РЗКТО складається з таких кроків:

1. Поступово (зліва направо та зверху вниз) при перегляді РЗКТО знайти перший неопрацьований контур (білого кольору).

2. Ініціалізувати лічильник вершин (ЛВ) вихідної триангуляційної моделі (ТМ) поверхні поточного ЛО. Під час розгляду нової вершини значення лічильника має збільшуватися на одиницю.

3. Задати рівень землі (РЗ) для поточного ЛО або для цілого РТЗО (може бути визначений автоматично на етапі виділення контурів ЛО).

4. Створити асоціативний список для збереження всіх порядкових номерів вершин ТМ ЛО з позиціями цих вершин у РЗТО. Для роботи запропонованого алгоритму асоціативний список вершин (АСВ) має виконувати над даними такі операції:

- додавати дані до списку;
- поступово переглядати всі елементи списку (порядок є не важливим);
- швидко знаходити порядковий номер вершини у ТМ ЛО за позицією цієї вершини у РЗТО.

Найкраще ці умови задовольняє інтерфейс карт відображень (map), що використовує хеш-таблицю. Мовою програмування Java 2 – це клас HashMap.

5. Позначити контур ЛО у РЗКТО певним кольором. Це дасть змогу надалі не аналізувати внутрішні контури ЛО. В процесі обробки кожної точки контуру (ТК) необхідно виконати такі додаткові операції:

- позицію точки контуру у РЗТО та присвоєний їй порядковий номер у ТМ ЛО (значення ЛВ) заносимо у АСВ;
- ЛВ та тривимірну координату ТК записуємо у файл триангуляції (ФТ);
- збільшуємо значення лічильника ЛВ на одиницю;
- якщо рівень ТК є вищим за заданий РЗ, то у ФТ додатково записуємо ЛВ та точку з двовимірними координатами контуру і РЗ як третьої координати. Позначаємо, що це додаткова точка РЗ та збільшуємо значення лічильника ЛВ на одиницю.

6. Побудувати бокові грані ТМ ЛО. Для цього для кожної ТК ЛО необхідно перевірити на рівність значення поточної та наступної за нею ТК із РЗ ЛО. За результатом цієї перевірки записуємо такі трикутники у ФТ (рис. 1):

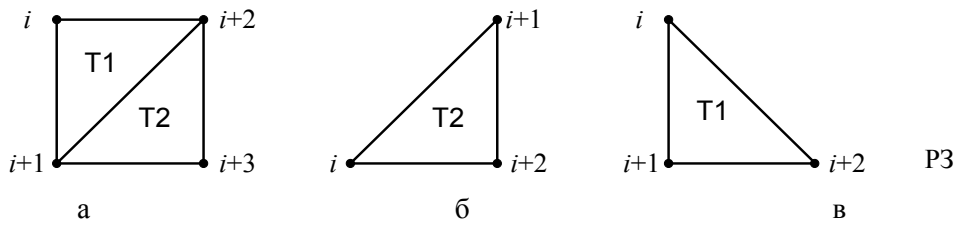


Рис. 1. Варіанти трикутників для поточної ТК в процесі побудови бокових граней ТМ ЛО:
 а – $TK_i \neq P3, TK_{i+2} \neq P3$; б – $TK_i = P3, TK_{i+1} \neq P3$; в – $TK_i \neq P3, TK_{i+2} = P3$

На рис. 1 символом i позначено порядковий номер поточної ТК у ТМ ЛО. Якщо результат перевірки не збігається з жодним із запропонованих варіантів на рис. 1, то переходимо до опрацювання наступної ТК. Для останньої ТК за наступну необхідно приймати першу ТК.

7. Виділити цілий ЛО обробки у РЗКТО шляхом його заливки певним кольором (відмінним від кольору контуру). Всі внутрішні контури при цьому також зафарбовуємо. Для цієї задачі можна використати простий та ефективний рекурсивний алгоритм заливки [14]. Як вхідні параметри алгоритм заливки потребує координати точки, яка є точкою об'єкта (ТО) та не є ТК. Оскільки перша знайдена ТК є найвищою ТО та є крайньою лівою на цій висоті, то за товщини контуру в один піксель можливі три варіанти продовження контуру від цієї точки, як показано на рис. 2.

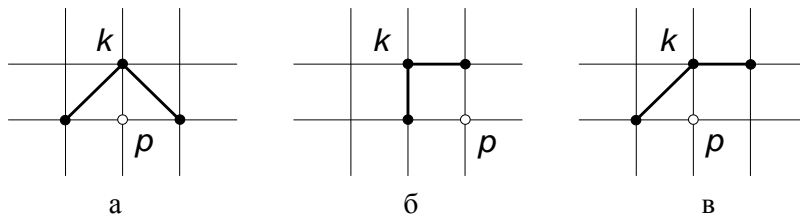


Рис. 2. Можливі варіанти продовження контуру від першої знайденої ТК та розташування ТО, яка не є ТК, у цих випадках

На рис. 2 точкою k позначено першу знайдену ТК, точкою p – ТО, яка не є ТК. В процесі виділення кожної нової ТО необхідно виконати такі додаткові операції:

- позицію ТО у РЗТО та присвоєний їй порядковий номер у ТМ ЛО (значення ЛВ) заносимо у АСВ;
- ЛВ та тривимірну координату ТО записуємо у ФТ;
- збільшуємо значення лічильника ЛВ на одиницю.

8. Побудувати ТМ ЛО. Для цього для кожної ТО ЛО необхідно перевірити на належність точок маски два на два пікселі ЛО, як це показано на рис. 3,а. За результатом цієї перевірки записуємо наступні трикутники у ФТ (рис. 3):

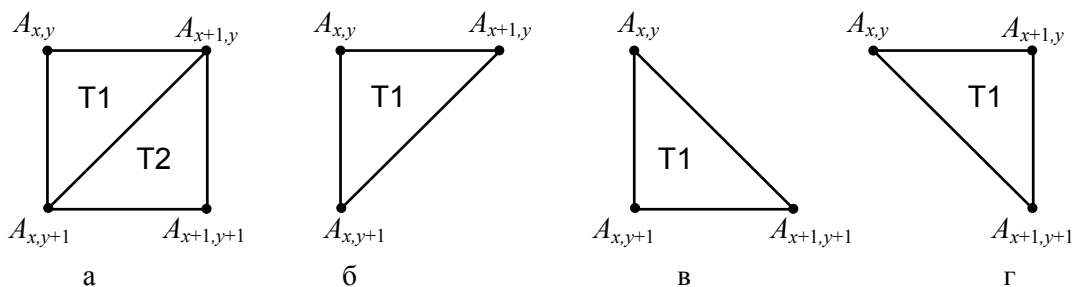


Рис. 3. Варіанти трикутників для поточної ТО в процесі побудови ТМ ЛО:

$$\begin{aligned}
 a - [(x, y), (x+1, y), (x+1, y+1), (x, y+1)] \in ПТО; & \quad b - [(x, y), (x+1, y), (x, y+1)] \in ПТО, (x+1, y+1) \notin \\
 & \quad ПТО; \quad в - [(x, y), (x+1, y+1), (x, y+1)] \in ПТО, (x+1, y) \notin ПТО; \\
 & \quad з - [(x, y), (x+1, y), (x+1, y+1)] \in ПТО, (x, y+1) \notin ПТО
 \end{aligned}$$

На рис. 3 значення $A_{x,y}$ відповідає порядковому номеру поточної ТО (x,y) у ТМ ЛО (визначається з АСВ). Якщо результат перевірки для певної ТО не збігається з жодним із варіантів на рис. 3, то переходимо до опрацювання наступної ТО. Однак, за таким підходом до побудови ТМ ЛО є випадок, коли трикутники ЛО залишаються неописаними, як це показано на рис. 4.

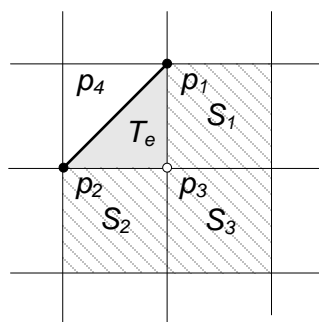


Рис. 4. Випадок неописаних трикутників у ТМ ЛО методом маркованих квадратів під час побудови непрямокутного ЛО

На рис. 4 точки p_1 та p_2 є ТК, а S_1 та S_2 відповідні цим точкам квадрати. Квадрат S_3 відповідає точці p_3 , яка є ТО, але не є ТК. Трикутник T_e залишився неописаним у ТМ ЛО, оскільки точка p_4 не є ТО. В запропонованому алгоритмі контур та об'єкт у РЗКТО виділяються різними кольорами. Це дає змогу легко визначити, до якої частини тривимірного об'єкта належить точка чи не належить взагалі. Тому перевіркою випадку (рис. 4) для кожної ТК можна легко вирішити цю проблему без значного збільшення складності алгоритму.

Запропонований алгоритм опису поверхні ЛО щільною регулярною ТС на РЗТО та РЗКТО, як і метод маркованих квадратів, має асимптотичну складність $O(N)$ операцій.

Фази алгоритму спрощення представлення поверхні кожного ЛО до заданого рівня деталізації та побудова тривимірної моделі території

Всі існуючі методи спрощення ТМ основані на одній з двох стратегій [11]:

- Стратегія “знизу–догори” починається з деталізованої триангуляції, згодом кількість елементів триангуляції поступово зменшується до досягнення необхідної кількості вузлів або поки не буде досягнуто заданого допустимого відхилення ε спрощеної триангуляції від вихідної.
- Стратегія “зверху–донизу” починається з простої апроксимуючої моделі, що складається з одного або декількох трикутників. Надалі до триангуляції послідовно додаються нові точки доти, поки не буде досягнуто необхідної точності.

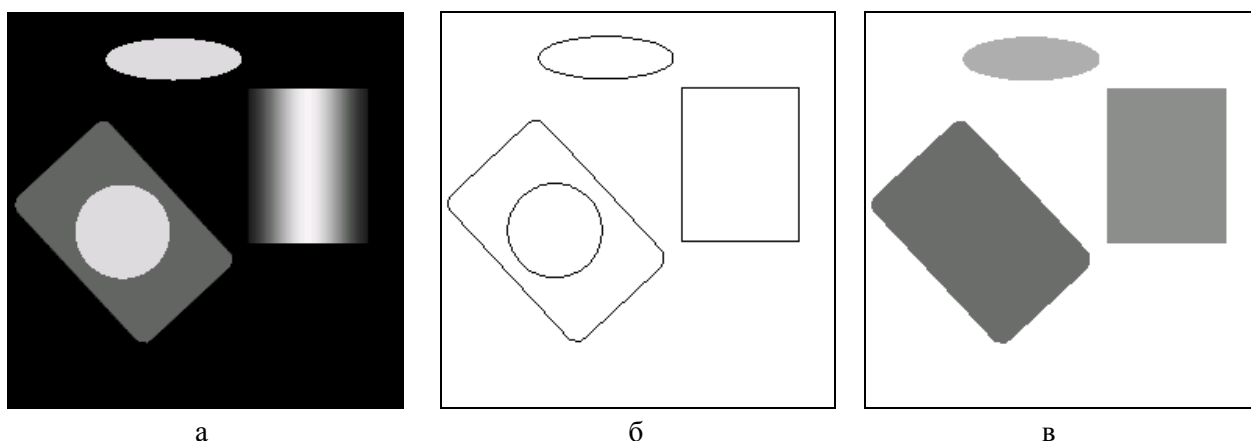
Так враховується взаємозалежність між точністю та стислістю необхідного подання вихідного об'єкта. Запропонований алгоритм автоматичної реконструкції поверхонь на РЗТО використовує стратегію “знизу–догори”. Ця стратегія дає змогу паралельно з процесом спрощення побудувати так звану ієрархію спрощень та зберегти її в спеціальній структурі – мультитриангуляції [11]. Мультитриангуляція дає можливість отримувати ТМ необхідної деталізації залежно від масштабу відображення та розташування ТМ до спостерігача, що має важливе практичне значення. Використання стратегії “знизу–догори” та алгоритму спрощення на основі механізмів прорідження (decimation) одночасно з спрощенням сітки згладить шуми на плоских ділянках поверхонь ЛО, що значно покращить якість відображення тривимірної моделі. До того за вибраною стратегією, як правило, процес відбувається швидше (асимптотична складність в середньому $O(N \log N)$).

У роботі використано алгоритм спрощення ТМ на основі механізмів прорідження [14]. Надалі всі ТМ ЛО об'єднуються в одне представлення. Ця операція є тривіальною.

2. Результати роботи алгоритму

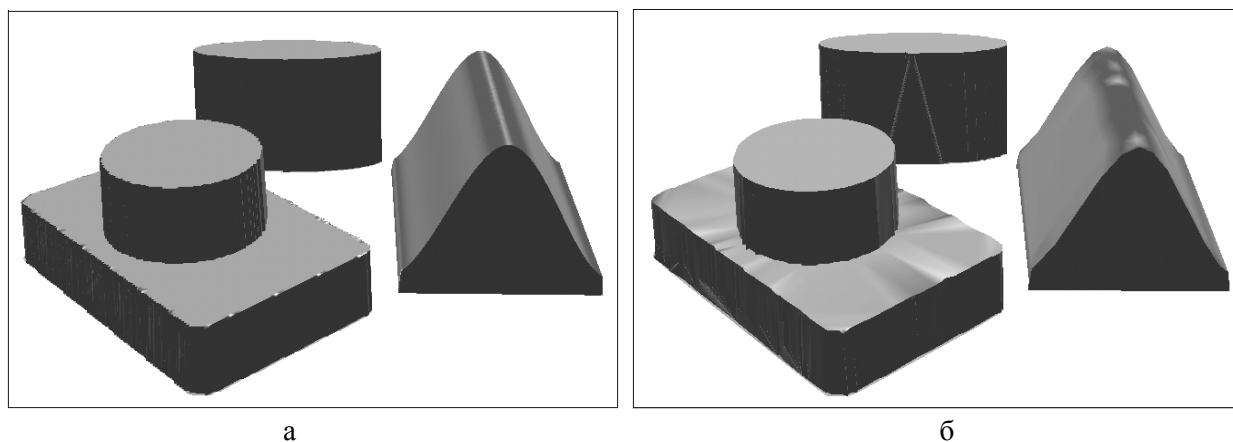
Приклад реалізації запропонованого методу на комп'ютері з процесором AMD Duron 1.4 ГГц та відеоадаптером NVIDIA GeForce2 MX 400 наведено на рис. 5–7.

На рис. 5, *а* показано тестовий приклад РЗТО та результати роботи алгоритмів виділення країв та зафарбовування ЛО на РЗТО. Тестовий приклад розміром 256×256 точок містить поверхні першого та другого порядку. З рис. 5, *б* видно, що алгоритм виділення країв формує контури замкнуті та потоншені до товщини контуру в один піксель. На рис. 5, *в* показано, що алгоритм зафарбовування ЛО позначає кожен ЛО різним кольором, причому колір ТК на одиницю більший за колір ТО, яка не є ТК. Всі внутрішні контури ЛО також зафарбовані.



*Рис. 5. Аналітичне описання країв об'єктів:
а – тестовий приклад; б – виділені контури; в – зафарбовані об'єкти*

На рис. 6 показано отриману тривимірну модель тестового РЗТО, подану щільною ТС (рис. 6, *а*) та спрощеною ТС (рис. 6, *б*). На рис. 7 показано фрагмент ТМ ЛО, що на рис. 7, *а* поданий щільною ТС, а на рис. 7, *б* – спрощеною ТС. Порівняльну характеристику ємності та часових затрат на створення і візуалізацію тривимірних моделей за типом їхнього представлення наведено у таблиці. Дані таблиці свідчать, що за незначних затрат на обробку значно зменшується час на візуалізацію з незначною втратою точності (рис. 6). До того ж спрощена тривимірна модель для збереження потребує значно менше пам'яті.



*Рис. 6. Тривимірна модель РЗТО:
а – подана щільною регулярною ТС; б – подана спрощеною ТС*

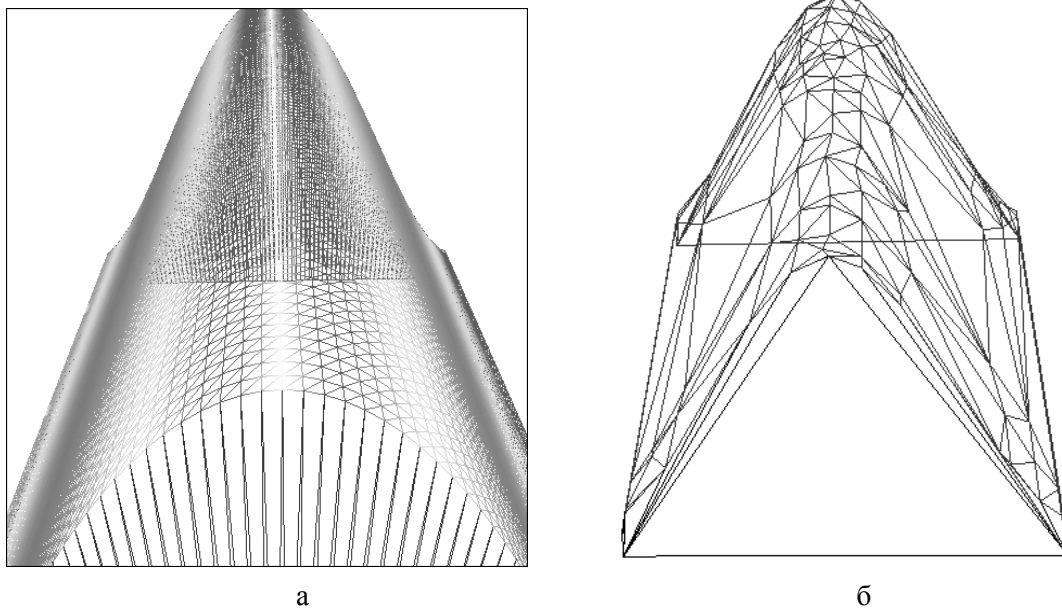


Рис. 7. Фрагмент ТМ ЛО:
а – поданий щільною ТС; б – поданий спрощеною ТС

Порівняльна характеристика ємності та часових затрат на створення і візуалізацію тривимірних моделей за типом їхнього представлення

Характеристика	РЗТО	Тривимірний модель, подана щільною ТС	Тривимірний модель, подана спрощеною ТС
Кількість вершин	65536	20755	851
Кількість трикутників	130050	40658	1626
Час створення ТМ поетапно, мс	1437	1844 (1 етап) 937 (2 етап)	1844 (1 етап) 937 (2 етап) 2612 (3 етап)
Загальний час створення ТМ, мс	1437	2781	5393
Час на візуалізацію тривимірної моделі, мс	25109	9094	2297
Загальний час на створення та візуалізацію, мс	26546	11875	7690

Висновки

1. Розглянуто новий підхід до вирішення проблеми автоматичної реконструкції поверхонь на РЗТО, який порівняно з відомими підходами використовує спрощену ТС для представлення форми ЛО та дає змогу описувати поверхні довільної форми.

2. Запропоновано процедуру відокремлення ЛО та алгоритм опису поверхні ЛО щільною регулярною ТС на РЗТО та РЗКТО на основі метода триангуляції маркованих квадратів з асимптотичною складністю $O(N)$ операцій на відміну від існуючих рішень на базі триангуляції Делоне зі складністю $O(N \log N)$ та $O(N^2)$.

3. Результати роботи алгоритму на тестових прикладах підтверджують ефективність запропонованого методу.

1. Vosselman G. Building reconstruction using planar faces in very high density height data // *www.citeseer.nj.nec.com/cs.* – 2000. – P. 6. 2. Haala N., Brenner C. and Anders K.-H. 3D urban GIS from laser altimeter and 2D map data. *International Archives of Photogrammetry and Remote Sensing.* – 1998. – Vol. 32, part 3/1. – P. 339–346. 3. Melnyk A., Emets V., Moroz I., Marchywka V. Reconstruction of 3-D objects from raster altimeter data // *The experience of designing and application of CAD systems in microelectronics / Proceeding of the VII International conference. CADSM 2003.* – Lviv, 18–22 February

2003. – P. 280–283. 4. Haithcoat T., Song W., Hipple J. Automated Building Extraction and Reconstruction from LIDAR Data // R&D Program for NASA/ICREST Studies Project Report. 09/16/01, 2001. – P. 27. 5. Rottensteiner F., Briese Ch. A new method for building extraction in urban areas from high-resolution LIDAR data // Institute of Photogrammetry and Remote Sensing, Vienna University of Technology. – 2003. – 7 p. 6. Morgan M., Habib A. Interpolation of LIDAR data and automatic building extraction // ACSM-ASPRS 2002 Annual conference proceedings. – 2002. – 10 p. 7. www.istar.com 8. Ємець В., Карпін О., Імад І.А. Сбієх. Система реконструкції тривимірних об'єктів за невпорядкованими даними аерозображень // Моделювання та інформаційні технології: Зб. наук. пр. – К., 2005. – Вип. 35. – С. 120–127. 9. Форсайт, Девід А., Понс, Жан. Комп'ютерне зречення. Современный подход: Пер. с англ. – М.: Изд. дом “Вильямс”, 2004. – 928 с.; Парал. тит. англ. 10. Ємець В.Ф., Мороз І.В. Алгоритм відокремлення країв на растрових зображеннях тривимірних об'єктів // Вісн. Нац. ун-ту “Львівська політехніка”. – 2003. – № 492. – С. 75–80. 11. Скворцов А. Триангуляція Делоне и её применение. – Томск: Изд-во Том. ун-та, 2002. – 128 с. 12. Skala V. Precision of iso-surface extraction from volume data and visualization // Proceedings of ALGORITHM 2000: Conference on Scientific Computing 2000. – 2000. – P. 368–378. 13. Эйнджер Э. Интерактивная компьютерная графика. Вводный курс на базе OpenGL. – 2-е изд.: Пер. з англ. – М.: Изд. дом “Вильямс”, 2001. – 592 с. 14. The MT (Multi-Tesselation) Package by Leila De Florian et al. // C++ library, University of Genova, <http://www.disi.unige.it/person/MagilloP/MT/>. – 2003.

УДК 681.3

Є.О. Кириченко, Н.Я. Павич*, Я.С. Парамуд, Ю.М. Спіченко**,
Національний університет “Львівська політехніка”
кафедра електронних обчислювальних машин,
*кафедра програмного забезпечення автоматизованих систем,
**проектно-конструкторське об'єднання “Політехніка”

ФАЙЛОВА СИСТЕМА СПЕЦІАЛІЗОВАНОГО ФЛЕШ-НАКОПИЧУВАЧА

© Кириченко Є.О., Павич Н.Я., Парамуд Я.С., Спіченко Ю., 2006

Розглянуто побудову файлової системи спеціалізованого флеш-накопичувача, який дає змогу приймати дані від різних джерел інформації, упаковувати їх у типові файли в реальному часі, обмінюватися з комп'ютером через інтерфейс USB.

The file system construction of specialized Flash-based Mass Storage Device is examined. The device allows taking information from various data sources, packing it into typical files in real-time mode, and exchanging them with computer through the standart USB interface.

Вступ

Одним із перспективних напрямів розвитку зовнішніх запам'ятовувальних пристроїв є широке впровадження флеш-накопичувачів [1]. Сучасний розвиток елементної бази для флеш-пам'яті [2–4] створює умови для побудови спеціалізованих флеш-накопичувачів. Актуальною прикладною задачею є приймання комп'ютером інформації від різних джерел для подальшого її опрацювання. У багатьох випадках джерела інформації використовують інтерфейси, через які безпосередньо, без спеціальних апаратно-програмних засобів, неможливо передати дані до універсального або персонального комп'ютера. Тоді доцільно використовувати спеціалізований флеш-накопичувач із вбудованою файловою системою, який забезпечував би узгодження різних