

УДК 004

МАТЕМАТИЧНІ МЕТОДИ ВІЗУАЛІЗАЦІЇ В 3D ПРОСТОРІ ЗА РЕЗУЛЬТАТАМИ ТОМОГРАФІЧНОГО ДОСЛІДЖЕННЯ ЗОБРАЖЕННЯ ЗЛОЯКІСНОЇ ПУХЛИНИ

Клейно О.Ю., Адамов В.Г.

Донецький національний технічний університет
кафедра Автоматизовані системи управління
E-mail: ok90@mail.ru, adamov@kita.dgtu.donetsk.ua

Анотація

Клейно О.Ю., Адамов В.Г. Математичні методи візуалізації в 3d просторі за результатами томографічного дослідження зображення злоякісної пухлини. Розглянуто математичні методи виділення контуру зображення, побудова тривимірної моделі складної геометричної поверхні шляхом апроксимації її поверхні трикутниками;

Загальна постановка проблеми

Метою розробки системи є виявлення патологічних змін утворень легенів, як при первинному виявленні, так і при наступному спостереженні за їхніми змінами. Це надалі значно спрощує порівняння даних, отриманих у різний час.

В даній роботі типовими задачами є:

- а) розпізнавання зображень за їх фотографіями;
- б) виділення контуру зображення;
- в) розрахунок об'ємів складних геометричних тіл;
- г) побудова тривимірної моделі складної геометричної поверхні шляхом апроксимації її поверхні трикутниками.

Початковими даними є чорно-білі фотографії-зрізи, отримані за допомогою комп'ютерного томографа.

Обрані математичні моделі повинні забезпечувати цифрову обробку вхідного зображення з виділенням його контуру, апроксимувати складну геометричну поверхню за допомогою трикутників, на основі введених даних щодо розміру фотографії та відстані між зрізами отримати значення об'єму отриманого тіла (пухлини)

Для цифрової обробки зображень на теперішній час використовується ряд відомих математичних методів [4].

Дамо аналітичний огляд існуючих математичних методів:

Методи виділення границь (контурів) або інакше виявлення границь працюють після того, як якість зображення поліпшена математичними методами обробки зображення.

Методи (алгоритми) виділення границь (контурів) можна розділити на два класи:

- а) методи, що підкреслюють, посилюють та виділяють границі;
- 2) методи, що будують контури, автоматично.

Відомі лінійні алгоритми виділення границь і нелінійні. У якості перших можна привести маски, схожі на ті, які використовуються в лінійній фільтрації. Різниця в тім, що для виділення границь використовуються вагові коефіцієнти різних знаків, як позитивні, так і негативні.

Наприклад, використання масок для горизонтальної або вертикальної обробки (виділення, посилення):

$$\begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix} \quad \text{та} \quad \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix}$$

В однотонних ділянках зображення, всі точки мають приблизно однакову яскравість, градієнт рівнів яскравості низький, і в результуючому зображенні подібні ділянки просто темніють. Там, де є перепади (різкі переходи, краї), крутість зміни яскравості висока і у кінцевому зображенні в таких місцях з'являються яскраві лінії.

Інший варіант - нелінійні алгоритми. До таким відносяться апертури Робертса та Собеля, Шарра, диференційні алгоритм

За результатом проведеного аналізу рекомендується застосовувати для виділення контуру об'єкту-зрізу оператор Собеля, як метод, що має найбільшу ефективність та швидкодію при достатній точності результатів.

Оператор Собеля дає досить якісний результат при обробці достатньо якісних зображень [3]. Точніше, оператор використовує значення інтенсивності тільки в околиці 3x3 кожного пікселя для одержання наближення відповідного градієнта зображення, і використовує тільки цілочисельні значення вагових коефіцієнтів яскравості для оцінки градієнта.

Формула оператора Собеля:

G_x і G_y - дві матриці, де кожна точка містить наближені похідні по x і по y . Вони обчислюються в такий спосіб шляхом множення матриці G_x і G_y і підсумовуванням обох матриць, у результаті отриманий результат записується в поточні координати x і y у нове зображення:

$$G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix} * A, \quad (1)$$

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} * A, \quad (2)$$

$$G = \sqrt{G_x^2 + G_y^2}. \quad (3)$$

Обґрунтування математичних методів вирішення задачі

Оператор обчислює градієнт яскравості зображення в кожній точці і таким чином визначається напрямком максимального збільшення яскравості. В цьому напрямку робиться перехід до наступної крапки, що лежить на границі. Результат показує, наскільки міняється яскравість зображення в кожній точці зображення та імовірність знаходження крапки на грані, а також орієнтація границі.

Математично, градієнт функції двох змінних для кожної точки зображення (якою і є функція яскравості) - двовимірний вектор, компонентами якого є похідні яскравості зображення по горизонталі і вертикалі. У кожній точці зображення градієнтний вектор орієнтований у напрямку максимального збільшення яскравості, а його довжина відповідає значенню зміни яскравості. Це означає, що результатом перетворення за оператором Собеля в точці області постійної яскравості буде нульовий вектор, а в точці, що лежить на границі областей різної яскравості - вектор, що перетинає границю в напрямку збільшення яскравості.

Нехай A вихідне зображення, а G_x і G_y - два зображення, де кожна точка містить наближені похідні по x і по y . Оскільки функція яскравості відома тільки в дискретних точках, ми не можемо визначити похідні доти, поки не покладемо яскравість безперервною функцією, що проходить через ці точки. Із цією додатковою передумовою похідну безперервної функції яскравості можна обчислити від функції, з якої взяті виміри - точки зображення. Виявляється, що похідні в будь-якій окремій точці є функції яскравості від всіх точок зображення. Однак наближення їх похідних можна визначити з достатнім ступенем точності.

Оператор Собеля можна реалізувати простими технічними та програмними засобами: для наближення вектора-градієнта потрібні тільки вісім пікселів навколо точка зображення та цілочисельна арифметика. Більше того, обоє дискретних фільтра можна розділити:

$$\begin{bmatrix} +1 & 0 & -1 \\ +2 & 0 & -2 \\ +1 & 0 & -1 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \cdot [+1 \quad 0 \quad -1] \quad (4)$$

та

$$\begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} \cdot [+1 \quad +2 \quad +1] \quad (5)$$

Тоді похідні – формули (1,2) можна обчислити за формулами:

$$G_x = \begin{bmatrix} +1 & 0 & -1 \\ +2 & 0 & -2 \\ +1 & 0 & -1 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} * ([+1 \quad 0 \quad -1] * A) \quad (6)$$

$$G_y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} * ([+1 \quad +2 \quad +1] * A) \quad (7)$$

що дозволить зменшити кількості арифметичних дій з кожним пікселем. Для визначення кута напрямку градієнта використовується формула:

$$\Theta = \operatorname{arctg}\left(\frac{G_y}{G_x}\right) \quad (8)$$

кут Θ дорівнює 0 для вертикальної границі, у якій темна сторона знаходиться зліва.

Програмна реалізація оператора Собеля може ефективно використати SIMD - розширення системи команд сучасних процесорів (так звана векторизація коду), при цьому вииграш у швидкості обчислення оператора може становити до 5 разів у порівнянні з високорівневою реалізацією.

Обчислення оператора Собеля елементарно розпаралелюється на довільне число потоків (у межі кожену точку результуючого зображення можна обчислювати незалежно від сусідніх). Наприклад, при наявності двох процесорів (ядер) верхній напівкадр зображення може бути оброблений одним з них, а нижній - іншим.

Застосуванням методу усіченого конусу, визначається об'єм отриманого складного геометричного тіла.

Для визначення об'єму усіченого конусу використовується формула:

$$V = \frac{1}{3} \pi \cdot h \cdot (r_1^2 + r_1 r_2 + r_2^2) \quad (9)$$

де r_1 та r_2 площі основ усіченого конусу (рис.1).

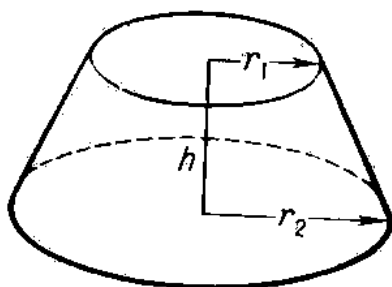


Рисунок 1 – До визначення об'єму усіченого конусу

Для побудови 3D – моделі складного геометричного об'єкту використовується метод триангуляції.

Генерація об'ємних зображень представляє складне обчислювальне завдання, у зв'язку із чим на практиці виконують її декомпозицію. Складні зображення формують фрагментарно, для чого їх розбивають на складові частини. Процес розбивки поверхні об'єктів на полігони одержав назву теселяції. Ця стадія на даному етапі розвитку машинної графіки проводиться повністю програмно поза залежністю від технічного рівня 3D-апаратури.

У цей час з'явилося велика розмаїтість графічних акселераторів, які мають апаратні засоби для зафарбування тривимірних об'єктів, видалення невидимих частин, накладення текстур і інші графічні функції. Для використання переваг 3D-прискорювачів необхідно програмно зробити теселяцію вихідних об'єктів, передаючи отримані полігональні області для подальшої обробки акселератором. [2]

При аналізі або синтезі складних поверхонь їх апроксимують сіткою трикутників, згодом чого надалі оперують із найпростішими полігональними областями.

Будь-яка поверхня може бути апроксимована з необхідною точністю сіткою трикутників. Точність апроксимації визначається кількістю трикутників і способом їхнього вибору.

Найпростіше рішення тріангуляції складається в розщепленні полігона уздовж деякої хорди на два полігони та надалі рекурсивній розбивці їх до ситуації, коли підлягаючої тріангуляції полігон є трикутником.

Даний спосіб можна застосовувати лише для тріангуляції опуклих полігонів. Опуклим вважається полігон, якщо відрізок прямої, що з'єднує будь-які його дві крапки, повністю лежить у внутрішній області (рис 2а). Полігон на малюнку 2б не є опуклим, тому що відрізок ab виходить за межі багатокутника.

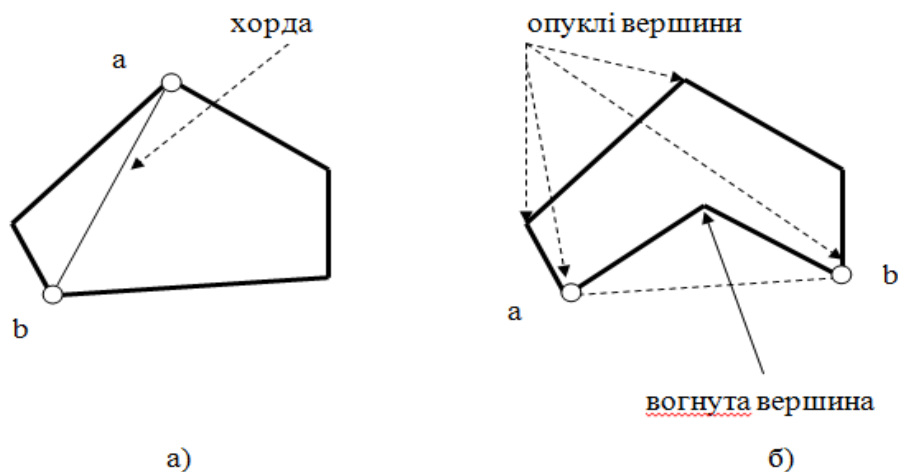


Рисунок 2 – Опуклі та не опуклі полігони.

Тріангуляція неопуклих полігонів не так проста, тому попередня розбивка неопуклих багатокутників на опуклі істотно спрощує алгоритми їхньої наступної обробки. Простіше всього це можна зробити шляхом послідовного переносу й повороту багатокутника так, щоб одна з його вершин збігалася з початком координат, а вихідна з неї сторона - з віссю OX . При розташуванні яких-небудь сторін нижче осі, відбувається їхнє відсікання, і алгоритм рекурсивно повторюють для отриманих полігонів, поки вони не стануть опуклими.

Тріангуляцію будь-якого полігона можна здійснити по наступному універсальному алгоритму.

Вибирається крайня ліва вершина й між двома її суміжними сторонами проводиться діагональ. При цьому можуть мати місце наступні два випадки: рис.3а - діагональ ac є хордою; рис.3б - діагональ ac не хорда, тому що усередину трикутника abc попадає вершина d багатокутника (у загальному випадку їх може бути декілька).

Із всіх вершин усередині трикутника abc вершина d найбільш віддалена від сторони ac . Цю вершину будемо називати тією, що вторгається. У випадку, коли вершини що вторгається немає, то отриманий трикутник заносять у сітку трикутників, і алгоритм рекурсивно обробляє полігон, що залишився, доти, поки він не виродиться в трикутник.

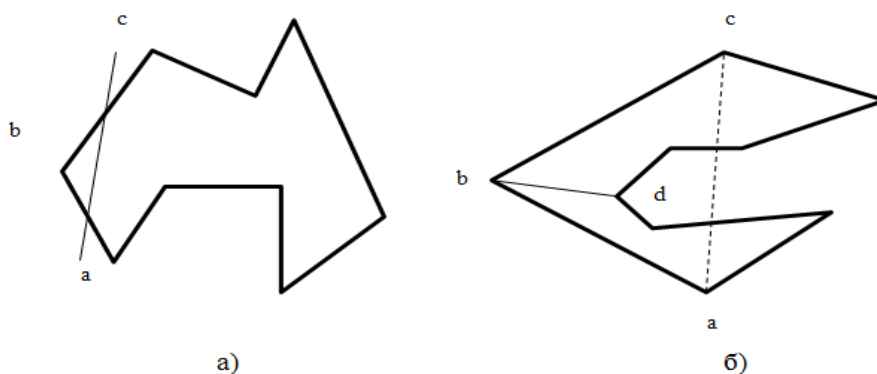


Рисунок 3 - Пояснення до універсального алгоритму триангуляції.

При виявленні вершини, що вторгається, проводиться діагональ із поточної до вершини, що вторгається. Отримані полігони знову рекурсивно обробляються до одержання трикутників.

У машинній графіці завдання триангуляції можна розглядати у двох напрямках - це триангуляція полігональних областей і триангуляції набору точок. Остання має місце при описі поверхні набором точок і інтенсивності їх кольорів.

Покрапковий опис поверхонь застосовують у тих випадках, коли поверхня дуже складна та не має гладкості, а детальне подання численних геометричних особливостей важливо для практики.

До поверхонь такого роду можна віднести пухлини та інші утворення зі складною формою, отримані за допомогою комп'ютерного томографа.

На основі отриманих алгоритмів в даній роботі в середовищі Visual C++ розроблена програма, що дозволяє виконувати виділення контуру об'єкту на томографічних знімках-зрізах; розрахунок об'єму виділеної області; побудову 3D-моделі пухлини.

Виконана перевірка працездатності програми по набору зрізів томографічних зображень злоякісної пухлини. Працездатність виявилася задовільною.

Висновки

В результаті виконаної роботи розроблено алгоритм і програма виділення контуру патологічної області, використовуючи, масиви томографічних зрізів. Проведено розрахунок параметрів, що характеризують пухлину (об'єм, щільність). На підставі триангуляції отримано алгоритм побудови 3D-моделі пухлин, що буде вбудовано в програмний комплекс.

Список літератури

1. Славин М.Б. «Системное моделирование патологических процессов» М. 1983.
2. Скворцов А.В. Триангуляция Делоне и её применение. — Томск: Изд-во Том. ун-та, 2002. — 128 с.
3. Тихомиров Ю. Программирование трехмерной графики. СПб., ВНУ 1998.
4. Хофер М. Компьютерная томография. Базовое руководство. 2-е издание, переработанное и дополненное: — М.: Мед.лит., 2008. — 224 с.