



УКРАЇНА

(19) UA

(11) 91329

(13) C2

(51) МПК (2009)

G02F 2/00

G06T 11/00

МІНІСТЕРСТВО ОСВІТИ  
І НАУКИ УКРАЇНИ

ДЕРЖАВНИЙ ДЕПАРТАМЕНТ  
ІНТЕЛЕКТУАЛЬНОЇ  
ВЛАСНОСТІ

## ОПИС ДО ПАТЕНТУ НА ВИНАХІД

**(54) СИСТЕМА І СПОСІБ НАНЕСЕННЯ ТРИВИМІРНИХ ОБ'ЄМНИХ ТЕКСТУР НА ДОВІЛЬНО ТРИАНГУЛЬОВАНІ ПОВЕРХНІ**

1

(21) а200602488  
(22) 05.08.2004  
(24) 26.07.2010  
(86) PCT/US2004/025298, 05.08.2004  
(31) 10/636,441  
(32) 06.08.2003  
(33) US  
(46) 26.07.2010, Бюл.№ 14, 2010 р.  
(72) СПАЙСЕР ШІН А., US  
(73) ЛЕНДМАРК ГРАФІКС КОРПОРЕЙШН, US  
(56) US 20020162081 A1; 31.10.2002  
US 20020060684 A1; 23.05.2002  
US 6438493 B1; 20.08.2002  
US 6483520 B1; 19.11.2002  
US 6064771 A; 16.05.2000  
(57) 1. Спосіб відображення графічних даних, що складаються принаймні з одної об'ємної інформаційної точки (102) в інформаційному об'ємі (100), який включає такі кроки:  
створення текстурного атласу, що складається з двовимірної схеми принаймні одного полігона (108) на полігональній поверхні;  
сканування-конвертація принаймні одного полігона (108) для утворення відповідного тексела (600); і  
запису кожного тексела, що перетинає полігональну поверхню, в сегмент пам'яті (500) для необов'язкового відображення.  
2. Спосіб за п. 1, в якому інформаційним об'ємом є сейсмічний інформаційний об'єм, що включає принаймні амплітуду, фазу, частоту, вигляд і когерентність.  
3. Спосіб за п. 2, що додатково включає крок створення текстурного елемента (300).  
4. Спосіб за п. 3, що додатково включає кроки: визначення, чи узгоджується принаймні один полігон (108) із заздалегідь визначеним максимальним розміром текстурного елемента; поділ принаймні одного полігона (108) на множину компонентних полігонів, якщо він не узгоджується з максимальним розміром текстурного елемента; і позначення принаймні одного полігона (108) і кожного з множини компонентних полігонів, що узгоджується з максимальним розміром текстурного елемента.  
5. Спосіб за п. 4, що додатково включає крок створення набору текстурних елементів, причому набір текстурних елементів включає множину текстурних елементів, множина текстурних елементів включає в себе тексту-

2

рний елемент, кожний з множини текстурних елементів включає принаймні один полігон (108) і множину компонентних полігонів.

6. Спосіб за п. 4, в якому принаймні один полігон (108), що не узгоджується з максимальним розміром текстурного елемента, має найдовшу грань, середню точку на найдовшій грані і вершину.

7. Спосіб за п. 6, який додатково включає такі кроки: створення грані між середньою точкою і вершиною; і виконання виродження центра ваги.

8. Спосіб за п. 5, що додатково включає такі кроки: визначення, чи узгоджується кожна множина текстурних елементів із заздалегідь визначеним максимальним розміром текстурного атласу; приписування до текстурного атласу кожної множини текстурних елементів, що узгоджується з максимальним розміром текстурного атласу; приписування до нового текстурного атласу кожної множини текстурних елементів, що не узгоджується з максимальним розміром текстурного атласу; і позначення кожної множини текстурних елементів, що приписана до текстурного атласу і нового текстурного атласу.

9. Спосіб за п. 8, що додатково включає крок порівняння розміру елемента для кожної множини текстурних елементів з максимальним розміром текстурного атласу, кожна множина текстурних елементів приписана до текстурного атласу і нового текстурного атласу, щоб звести до мінімуму різницю між розміром колективного елемента для множини текстурних елементів, приписаних текстурному атласу, і максимальним розміром текстурного елемента, і щоб звести до мінімуму різницю між розміром колективного елемента для множини текстурних елементів, приписаних до нового текстурного атласу, і максимальним розміром текстурного атласу.

10. Спосіб за п. 1, в якому тексел (600) базується на відповідній об'ємній інформаційній точці (102).

11. Спосіб за п. 1, який додатково включає крок обробки представлення принаймні одного полігона (108) в якому оброблене представлення принаймні одного полігона (108) включає принаймні одну вершину для полігона.

12. Спосіб за п. 1, що додатково включає крок відображення принаймні частини перетину тексела (600) і полігональної поверхні.

(13) C2

(11) 91329

(19) UA

13. Система для відображення графічних даних, що включає принаймні одну об'ємну інформаційну точку в інформаційному об'ємі (100), пристрій для зберігання програми, що зчитується машиною, який містить програму з інструкціями, що виконуються машиною для здійснення наступних кроків: створення текстурного атласу, який складається з двовимірної схеми принаймні одного полігона (108) на полігональній поверхні; сканування-конвертація принаймні одного полігона (108) для утворення відповідного тексела (600); і запису кожного тексела, що перетинає полігональну поверхню, в сегмент пам'яті (500) для неов'язкового відображення.

14. Система за п. 13, в якій інформаційний об'єм - це сейсмічний інформаційний об'єм, що включає принаймні один атрибут з: амплітуди, фази, частоти, вигляду і когерентності.

15. Система за п. 13, в якій текстурний атлас включає текстурний елемент (300).

16. Система за п. 15, яка додатково містить крок створення набору текстурних елементів, який містить текстурний елемент (300) та інший текстурний елемент.

17. Система за п. 16, в якій текстурний атлас містить принаймні один з текстурних елементів (300) та інший текстурний елемент.

18. Система за п. 17, що далі складається з кроку створення набору текстурних атласів, набір текстурних атласів складається з текстурного атласу та іншого текстурного атласу, інший текстурний атлас містить принаймні один текстурний елемент (300) та інший текстурний елемент.

19. Система за п. 15, в якій текстурний елемент (300) містить полігон.

20. Система за п. 13, в якій тексел (600) базується на відповідній об'ємній інформаційній точці.

21. Система за п. 13, яка додатково передбачає крок обробки представлення принаймні одного полігона (108), в якій оброблене представлення принаймні одного полігона (108) містить принаймні одну вершину для полігона.

22. Система для нанесення тривимірної текстури на полігональну поверхню, що містить комп'ютерно-зчитуваний носій, що має структуру даних, яка зберігається в ньому у вигляді текстурного атласу, який містить:

засоби для генерування текстурного елемента (300), що містять:

окреме число, що відрізняє текстурний елемент (300) від інших текстурних елементів;

координати, що визначають положення текстурного елемента (300) на текстурному атласі;

засоби для генерування розміру текстурного атласу;

засоби для генерування співвідношення розмірів текстурного атласу; і

засоби для генерування координатів, що визначають положення текстурного атласу в сегменті пам'яті (500), призначеному для текстурного атласу.

23. Система за п. 22, в якій текстурний елемент (300) містить:

представлення полігона, яке має вершини, причому вершини принаймні частково містяться у тривимірному інформаційному об'ємі;

число, що визначає полігон;

розмір елемента;

число, що визначає текстурний атлас, що містить елемент; і

координати, що визначають положення елемента на відображеному зображенні.

24. Система за п. 23, в якій інформаційний об'єм містить сейсмічні дані, що включають принаймні один атрибут з: амплітуди, фази, частоти, вигляду і когерентності.

25. Система за п. 23, в якій полігон є трикутником.

26. Система за п. 23, в якій представлення полігона складається з окремих координат для кожної з його вершин.

27. Система за п. 22, в якій сегмент пам'яті (500) містить текстурний кеш.

Цей винахід пов'язаний з системами і способами відображення графічних даних, зокрема з способами нанесення точних тривимірних об'ємних текстур на довільно триангульовані поверхні.

В певних областях корисно моделювати реальні чи створені комп'ютером об'єкти в трьох вимірах. Моделювання таких об'єктів виявилось корисним у багатьох галузях застосування. Наприклад, моделювання підземної структури ділянки земної кори є корисним для виявлення родовищ, визначення положення ліній скидання та в інших геологічних галузях застосування. Подібним чином, моделювання частин тіла людини є корисним для медичного навчання, діагностики, виконання дистанційних операцій чи для інших медичних галузей застосування. Моделювання створених комп'ютером об'єктів є корисним у розробці комп'ютерних ігор, проектування імітації польотів та інші галузі застосування. Існують також інші галузі застосування тривимірного моделювання реальних чи створених комп'ютером об'єктів.

Певні тривимірні моделі представляють об'єкт у вигляді тривимірної матриці об'ємних інформаційних точок. Така матриця, відома як інформаційний об'єм, містить множину інформаційних даних, відомих як об'ємні інформаційні дані. Кожні об'ємні інформаційні дані можна пов'язати з об'ємним пікселем, відомим також як воксел. Воксел - це найменша помітна кубічна частина тривимірного зображення. Воксел подібний пікселу, проте представляє тривимірний об'єм, а не тривимірну область. Кожний воксел представляє дискретну вибірку тривимірної частини модельованого об'єкта.

Кожний воксел в інформаційному об'ємі містить унікальний набір координат (x, y, z) і одне чи більше значень інформаційних даних, що представляють певну властивість, що досліджується. Ці значення даних можна показати на шкалі (0-256), яка відповідає певним кодам червоного, зеленого, блакитного кольору (RGB) і змінній непрозорості (A), які одержані з результату вимірювання чи значення властивості, що досліджується. Ре-

зультат вимірювання чи значення властивості відповідає частині об'єкта, який представляє воксел. Це дає можливість графічно представити властивість, що досліджується.

Наприклад, в сейсмічних чи геологічних галузях застосування, сейсмічні дані одержують шляхом штучного створення звукових хвиль і запису появи цих звукових хвиль після відбиття від поверхонь поділу підземних скелястих формацій з різними фізичними властивостями. Потім з відбитих звукових хвиль створюють об'ємні дані, що представляють підземну структуру, чи сейсмічні дані. В такому прикладі кожний воксел містить значення даних, що представляє хвилю, що проходить (тобто, амплітуду), з відбитої звукової хвилі. З відображеної картини цього значення даних можна зробити висновок про геологічну структуру. Наприклад, найвищі значення даних сейсмічної амплітуди можуть бути представлені червоним кольором, трохи нижчі значення даних амплітуди - жовтогарячим кольором, а інші значення даних амплітуди - іншими кольорами. Можливі інші кольорові схеми. Це дає можливість представити сейсмічні дані.

Після створення інформаційного об'єму його вміст можна відобразити для користувачів. Користувач може визначити тривимірну поверхню, що міститься в об'ємі, а система відображення відобразить вокселі на цій поверхні на екрані чи на іншому середовищі відображення. Це дає можливість користувачу переглядати вокселі, що містяться всередині інформаційного об'єму.

Одна з систем відображення вокселів на довільній тривимірній поверхні відома як тривимірне текстурне відображення. Цей спосіб здійснюється у пристрої растрової обробки апаратного графічного прискорювача комп'ютера. Цей спосіб використовує певну спеціалізовану пам'ять, відому як текстурний кеш чи текстурна пам'ять, що встановлюється окремо на інтегральній мікросхемі.

У цьому способі текстурний кеш використовується у якості буфера для інформації, що міститься в інформаційному об'ємі. Блок чи частина цього інформаційного об'єму зчитується в текстурний кеш, а інформація з цього блоку використовується пристроєм растрової обробки. Потім, другий блок інформаційного об'єму зчитується в текстурний кеш, а інформація з цього блоку використовується пристроєм растрової обробки. Цей процес продовжується до тих пір, поки весь інформаційний об'єм не надійде до пристрою растрової обробки, по одному блоку за раз, через текстурний кеш. Таким способом текстурний кеш використовується як буфер для інформаційного об'єму.

Такий спосіб має певні власні властивості, що роблять його неефективним у деяких ситуаціях. Наприклад, якщо інформаційний об'єм перевищує розмір наявного текстурного кеша, то інформаційний об'єм необхідно розбити на блоки, а кожний блок слід перекачати в текстурний кеш. Це неефективно через відносно великі накладні витрати, потрібні у визначенні розміру кожного блоку і зчитуванні його у текстурний кеш.

Крім того, хоча в такому способі пристрою растрової обробки можуть знадобитися лише вокселі, що лежать у тривимірній поверхні, кожний вок-

сел в інформаційному об'ємі зчитується в текстурний кеш і зберігається в ньому. Таким чином, пам'ять в текстурному кеші використовується неефективно через відносно велику кількість непотрібних даних, які можуть зберігатися в текстурному кеші. Ця проблема загострюється, коли тривимірна поверхня є невеликою у порівнянні з інформаційним об'ємом.

Крім того, оскільки у цій операції використовується відносно великий об'єм текстурного кеша, то об'єм текстурного кеша для інших операцій з графікою є обмеженим. Це обмежує продуктивність рендерингу об'єкта та інших операцій з графікою.

Для відображення вмісту інформаційного об'єму на довільній тривимірній поверхні, що міститься в об'ємі, серед іншого, потрібен спосіб, який скоротив би об'єм текстурного кеша.

В одній особливості здійснення цей винахід вирішує ці та інші проблеми у області техніки, пов'язаній з нанесенням тривимірної об'ємної текстури на полігон. Цей спосіб включає в себе кроки забезпечення інформаційного об'єму, що складається з множини об'ємних інформаційних точок, забезпечення полігональної поверхні, що частково чи повністю міститься в інформаційному об'ємі, полігональної поверхні, що складається принаймні з одного полігона, створення набору текстурних атласів, кожний з яких містить двовимірну схему принаймні одного полігона на полігональній поверхні, і виділення сегмента пам'яті для кожного текстурного атласу. Цей спосіб також включає в себе кроки сканування-конвертації кожного полігона, що ґрунтується на об'ємних інформаційних даних в інформаційному об'ємі, і запис результатів цього сканування-конвертації кожного полігона в сегмент пам'яті, виділеної для текстурного атласу, що містить схему з цим полігоном.

В іншій особливості, даний винахід включає в себе текстурний елемент з вимірами, що описують розмір цього елемента і вигляд полігона, при цьому полігон повністю чи частково міститься у тривимірному інформаційному об'ємі.

Ще в іншій особливості, текстурний атлас включає в себе виміри атласу, що описують розмір цього атласу і вигляд полігона, при цьому полігон повністю чи частково міститься у тривимірному інформаційному об'ємі.

У ще одній додатковій особливості, сегмент пам'яті містить множину текселів (базові елементи текстурованої графіки), множину текселів, включаючи один чи більше заздалегідь визначених полігональних текселів і один чи більше заздалегідь визначених неоплігональних текселів, при цьому кожний полігональний тексел в множині текселів містить значення, обчислене на основі об'ємної інформаційної точки в інформаційному об'ємі, а кожний інший тексел є недійсним.

Ще в іншій особливості, даний винахід включає в себе систему для нанесення тривимірної об'ємної текстури на полігональну поверхню. Ця система включає в себе текстурний кеш, конфігурований таким чином, щоб містити множину текселів, і пам'ять, конфігуровану таким чином, щоб містити полігональну поверхню і інформаційний об'єм, полігональну поверхню, що складається з

одного чи більше полігонів, і інформаційний об'єм, що складається з однієї чи більше об'ємних інформаційних точок. Ця система також включає в себе керівний модуль, конфігурований таким чином, щоб створювати набір текстурних атласів, при цьому кожний текстурний атлас містить двовимірну схему принаймні одного полігона на полігональній поверхні, виділяти сегмент текстурного кеша для кожного текстурного атласу, сканувати-конвертувати принаймні один полігон, створюючи таким чином один чи більше текселів, і записувати один чи більше текселів, одержаних в результаті цього сканування-конвертації кожного полігона в сегмент текстурного кеша, виділеного для текстурного атласу, що містить схему з цим полігоном.

Даний винахід буде описаний з посиланням на супровідні графічні матеріали, в яких аналогічні елементи позначаються аналогічними номерами, і в яких:

На Фіг.1 показаний інформаційний об'єм і тривимірна поверхня.

На Фіг.2 показана блок-схема, що відображає спосіб нанесення інформаційних даних на поверхню.

На Фіг.3 показаний набір текстурних елементів.

На Фіг.4 показаний набір текстурних атласів.

На Фіг.5 показаний сегмент текстурної пам'яті.

На Фіг.6 показаний полігон до і після сканування-конвертації.

На Фіг.7 показана блок-схема системи для нанесення тривимірної об'ємної текстури на полігональну поверхню.

На Фіг.8 показана блок-схема, що відображає спосіб поділу полігона і створення текстурного елемента.

На Фіг.9 показана блок-схема, що відображає спосіб створення текстурного атласу.

На Фіг.1 показаний інформаційний об'єм і тривимірна поверхня у відповідності до здійснення даного винаходу. Інформаційний об'єм 100 може представляти собою чи включати в себе тривимірну модель реального чи створеного комп'ютером об'єкта. Інформаційний об'єм 100 може представляти собою чи включати в себе тривимірну матрицю об'ємних інформаційних точок 102a-d.

Кожна з об'ємних інформаційних точок 102a-d може містити одне чи більше значень, що описують певну частину інформаційного об'єму 100. Наприклад, кожна з об'ємних інформаційних точок 102a-d може містити координати  $x$ ,  $y$ ,  $z$ , що визначають положення частини інформаційного об'єму 100. Кожна з об'ємних інформаційних точок 102a-d може містити також одне чи більше значень, що відповідають одній чи більше певним властивостям. Наприклад, кожна з об'ємних інформаційних точок 102a-d може містити значення даних, що відповідає сейсмічній властивості, густині чи певній іншій властивості.

В одному з прикладів здійснення, кожна з об'ємних інформаційних точок 102a-d може представляти собою чи включати в себе воксел. Кожний воксел можна відобразити як комбінований код кольору і змінну непрозорості (RGBA), що оде-

ржани з вимірювання чи значення певної властивості.

Інформаційний об'єм 100 можна визначити у будь-якій системі координат, наприклад, у декартовій системі координат з початком координат 104.

Полігональна поверхня 106 може повністю чи частково міститися в інформаційному об'ємі 100. Полігональна поверхня 106 може бути дво- чи тривимірною поверхнею, що вибрана чи визначена користувачем чи комп'ютером. Полігональна поверхня 106 може бути вибрана чи визначена користувачем для дослідження об'ємних інформаційних точок 102a-d, що перетинають полігональну поверхню 106. В деяких галузях застосування полігональна поверхня 106 може бути вибрана для того, щоб точно при приблизно моделювати природну поверхню. Наприклад, полігональна поверхня 106 може бути вибрана для того, щоб моделювати поверхню поділу між шарами земної кори.

Полігональна поверхня 106 може бути визначена одним чи більше полігонами 108a-d. Для опису кожного полігона можна застосувати різні способи. Наприклад, кожний полігон можна визначити набором вершин і кутів. В одному прикладі здійснення, полігони 108a-d є трикутниками, і кожний з полігонів 108a-d визначається декартовими координатами своїх вершин.

На Фіг.2 показана блок-схема, що відображає спосіб нанесення інформаційних даних на поверхню у відповідності до здійснення даного винаходу. Застосування цього способу розпочинається з кроку 200, в якому можна одержати інформаційний об'єм 100. Інформаційний об'єм 100 можна створити з вимірних даних, таких як сейсмічні дані, чи створити їх іншим чином.

У кроці 202 можна одержати полігональну поверхню 106. Полігональна поверхня 106 може бути визначена користувачем чи комп'ютером. Полігональна поверхня 106 може бути вибрана для того, щоб дослідити об'ємні інформаційні точки 102a-d, що перетинають полігональну поверхню 106. Полігональна поверхня 106 може бути вибрана для того, щоб точно чи приблизно моделювати природну поверхню, чи може бути вибрана іншим чином.

У кроці 204 стиснута індексація полігонів 108a-d може бути видалена. В деяких галузях застосування, полігони 108a-d можуть бути представлені у формі стиснутої індексації, відомої також як вершинна індексація чи видалення трикутників. Наприклад, кожна з вершин 110a-d представляє собою вершину двох чи більше полігонів. Для того щоб видалити цю форму стиснутої індексації і призначити набір вершин, унікальних для кожного полігона для окремої ідентифікації, кожному полігону 110a-d призначається номер вершини  $V_n$  і номер полігона  $P_n$ . Таким способом кожний полігон можна окремо індексувати у відповідності до його власного набору вершин. Наприклад, полігон, зв'язаний 110a-c, буде представлений  $V_n=0,1,2$  і  $P_n=0$ ; полігон, зв'язаний 110b-d, буде представлений  $V_n=0,1,2$  і  $P_n=1$ ; а полігон, зв'язаний 110c-e, буде представлений  $V_n=0,1,2$  і  $P_n=2$ . У цій області техніки можуть бути відомими інші способи вида-

лення цієї та інших форм стиснутого індексування, тому вони не будуть повторюватися тут.

У кроці 206 полігони 108a-d можна розділити. Цей крок може забезпечити, що кожний полігон 108a-d буде уміщатися в текстурний елемент у кроці 208. В одному з прикладів здійснення, визначається максимальний розмір текстурного елемента. У цьому прикладі здійснення деякі полігони можуть не уміщатися у найбільший допустимий текстурний елемент. У цьому випадку ці полігони можна розділити на менші полігони, що уміщуються в текстурний елемент допустимого розміру.

У кроці 208 текстурний елемент можна створити для кожного полігона 108a-d. Текстурні елементи, створені у цьому кроці, описані з посилання на Фіг.3. Кожний текстурний елемент 300a-d може включати в себе чи представляти собою двовимірний масив чи іншу частину електронної пам'яті, що має розмір відповідно до певного полігона 108a-d. Якщо певний полігон 108a-d є меншим, ніж заздалегідь визначений розмір елемента, то, в одному з прикладів здійснення, кожний текстурний елемент 300a-d включає в себе найменший двовимірний масив, що містить певний полігон 108a-d. В прикладі здійснення, де певний полігон 108a-d є меншим, ніж заздалегідь визначений розмір елемента, кожний текстурний елемент 300a-d включає в себе виміри масиву, що є достатньо великим, щоб містити певний полігон 108a-d, проте не містити сам фізичний масив.

Кожний текстурний елемент 300a-d може містити інформацію, що описує певний відповідний полігон 108a-d. Наприклад, кожний текстурний елемент може складатися з координат (x,y,z) вершин для кожного відповідного полігона 108a-d і номера (Pn), який ідентифікує цей полігон.

Кожний текстурний елемент 300a-d може також містити номер (Tn), що ідентифікує цей елемент, номер (An), що ідентифікує текстурний атлас, створений у кроці 210 і пов'язаний з цим елементом, координати (s',t'), що ідентифікують положення цього елемента в текстурному атласі, координати (s,t), що ідентифікують положення цього елемента у відображеному зображенні, і координати (x',y'), що ідентифікують початок координат для полігональної поверхні 106.

У кроці 210, текстурні елементи 300a-d можуть комбінуватися для утворення одного чи більше текстурних атласів. Створені у цьому кроці текстурні атласи описані з посиланням на Фіг.4. Кожний текстурний атлас 400a-c може представляти собою чи включати в себе двовимірну схему текстурних елементів 300a-d. Кожний текстурний елемент 300a-d можна помістити в один текстурний атлас 400a-c. Якщо в жодному існуючому текстурному атласі 400a-c немає місця, то можна створити новий текстурний атлас 400a-c. В певних прикладах здійснення, користувач може визначити цільовий чи максимальний розмір кожного текстурного атласу 400a-c.

В іншому прикладі здійснення кожний текстурний атлас 400a-c може представляти собою чи включати в себе двовимірний масив чи інший тип електронної пам'яті, конфігурованої таким чином, щоб містити один чи більше текстурних елементів

300a-d. Ще в одному прикладі здійснення кожний текстурний атлас 400a-c може містити інформацію про те, яким чином текстурні елементи 300a-d можна оптимально розташувати у двовимірній схемі, включаючи положення кожного текстурного атласу 400a-c в сегменті пам'яті 500, розглянутому з посиланням на крок 212. У цьому прикладі здійснення може виявитися непотрібним, щоб текстурні атласи мали достатню фізичну пам'ять, щоб містити кожний текстурний елемент 300a-d.

Як вже описувалося, створення текстурних елементів 300a-d в кроках 206 і 208 і створення текстурних атласів 400a-c у кроці 210 можна виконувати оптимальним чином. Це скоротить чи зведе до мінімуму порожнє чи невикористане місце в текстурних елементах 300a-d, текстурних атласах 400a-c, чи в обох цих місцях. Таким способом наявне місце текстурної пам'яті оптимізується за допомогою вищезгаданих способів. Можна також застосовувати інші добре відомі способи оптимізації, такі як способи, описані в роботі Баттаке, Шталлінга і Хеге Швидка лінійна інтегральна конволюція для довільних поверхонь в 3D у виданні 1997 року Візуалізація і математика.

У кроці 212 можна виділити сегмент пам'яті для кожного текстурного атласу 400a-c. Сегменти пам'яті, виділені у цьому кроці, описані з посиланням на Фіг.5. Розміри сегмента пам'яті 500, виділеного для певного текстурного атласу 400a, можуть бути рівними чи пропорційними розмірам текстурного атласу 400a. Сегмент пам'яті 500 може представляти собою чи включати в себе один чи більше сегментів текстурної пам'яті чи текстурного кеша. Сегменти пам'яті 500 можуть містити множину текселів 502a-d. Тексел - це базовий елемент текстурованої графіки, графіки, що визначає поверхню тривимірного об'єкта. Текселі 502a-d можуть бути неподільними елементами текстури, присутньої в сегменті пам'яті 500. Один чи більше текселів 502a і 502b можуть відповідати частині текстурного атласу 400a, що повністю чи частково міститься у полігоні 108d. Ці текселі 502a і 502b можна назвати полігонними текселями. Один чи більше текселів 502c можуть відповідати частині текстурного атласу 400a, що міститься у текстурному елементі 300d, проте не міститься в будь-якому полігоні 108a-d. Ці текселі можна назвати елементними текселями. Один чи більше текселів 502d, що не є ні полігонними текселями, ні елементними текселями, можна включити в сегмент пам'яті 500. Оскільки інформацію, що міститься в текстурному атласі і текстурних елементах, можна використати для визначення того, які текселі в сегменті пам'яті є полігонними текселями, то полігонні текселі в сегменті пам'яті 500 можна вважати заздалегідь визначеними.

У кроці 214 кожний полігон 108a-d можна сканувати-конвертувати. Крок сканування-конвертації полігона описаний з посиланням на Фіг.6. Сканування-конвертація - це процес, за допомогою якого значення даних, що містяться в інформаційному об'ємі 100, перетворюються в текселі. Одна чи більше об'ємних інформаційних точок 102a-d в інформаційному об'ємі 100 можуть перетинати певний полігон 108d. З інформації, що міститься в

об'ємних інформаційних точках 102a-d, що перетинають полігон, створюються текселі 600a-d, що відповідають об'ємним інформаційним точкам 102a-d. Якщо об'ємні інформаційні точки 102a-d включають вокселі, то ці вокселі можна перетворити у відповідні текселі 600a-d за допомогою добре відомого математичного способу, що називається трилінійною інтерполяцією. За допомогою цього способу кожний тексел визначається шляхом лінійного зважування восьми (8) вокселів, що є найближчими до тексела у інформаційному об'ємі 100. Текселі 600a-d, що створені шляхом сканування-конвертації, можуть відповідати полігонним текселям 502a-d, що містяться в сегменті пам'яті 500. Розміщення текстурних елементів в кожному текстурному атласі і положення кожного текстурного атласу в сегменті пам'яті не встановлюється до цього кроку.

У кроці 216 результати сканування-конвертації можна записати в сегмент пам'яті 500. Це може включати в себе запис текселів 600a-d в полігонні текселі 502a-d, в сегменті пам'яті 500. В одному з прикладів здійснення кожний тексел 600a-d можна записати у відповідний полігонний тексел 502a-d, коли він створений. В іншому прикладі здійснення можна створити текселі 600a-d, що відповідають певному полігону 108a-d, а усі текселі 600a-d для полігона 108a-d можна записати в сегмент пам'яті 500. В іншому прикладі здійснення можна створити текселі 600a-d, що відповідають певному текстурному атласу 400a-c, і записати їх як групу, чи можна створити усі текселі, що відповідають усім полігонам 108a-d, і записати їх як групу.

Сегмент пам'яті 500 може містити інформацію, корисну для рендерингу поверхні 106. Отже, ці текселі можна вважати недійсними. В іншому прикладі здійснення недійсними текселями 502c-d можуть бути порожні текселі, текселі, що містять лише початкові значення, чи текселі, що позначені недійсними іншим чином. В іншому прикладі здійснення недійсні текселі 502c-d можуть містити тексельні дані, наприклад, тексельні дані з попереднього рендерингу поверхні чи іншої операції.

Після запису усіх полігонних текселів 502a-b для усіх полігонів на полігональній поверхні 106 в сегменті пам'яті 500, сегменті пам'яті 500 можуть містити потрібну інформацію для відображення інформації, що міститься в інформаційному об'ємі 100 на поверхні 106.

У кроці 218 можна відобразити результати для користувача. Вміст сегментів пам'яті 500 можна відобразити за допомогою електронно-променевої трубки чи іншого екрану, їх можна роздрукувати чи відобразити іншим чином.

Цей спосіб можна застосовувати в сполученні з іншими системами і способами для графічного відображення інформаційного об'єму. Наприклад, цей спосіб можна застосувати в сполученні з системою аналізу і відображення наборів тривимірного інформаційного об'єму, описаних у спільній патентній заявці США "Система і спосіб для аналізу і відображення наборів тривимірних інформаційних об'ємів", серійний номер 09/119,635. При практичному застосуванні даного винаходу в сполученні з таким пристроєм, користувач може інтерактивно

створювати, формувати, масштабувати і переміщувати ряд вибірових проб, чи визначених користувачем поверхонь, в усьому наборі тривимірних інформаційних об'ємів. Перетин вибіркової проби з усім набором тривимірних інформаційних об'ємів є текстурою, відображеною на поверхнях вибіркової проби, чи об'єм, рендерований із змінними мірами прозорості у вибірковій пробі. Наприклад, коли користувач переміщує вибірову пробу, "клацаючи і перетаскуючи мишею", то він сприймає зміну текстури а поверхні вибіркової проби в "реальному масштабі часу" з переміщенням вибіркової проби. Таким способом користувач може інтерактивно переміщувати вибірові проби через весь тривимірний об'єм і легше та ефективніше візуалізувати і інтерпретувати властивості і фізичні параметри, що присутні в географічному просторі, представленому усім набором тривимірних інформаційних об'ємів.

Тепер повернемося до Фіг.7, на якому у відповідності до здійснення даного винаходу показана блок-схема системи нанесення тривимірної об'ємної текстури на полігональну поверхню. Система 700 може включати в себе текстурний кеш 702, що містить множину текселів 704. Система 700 також включає в себе пам'ять 706, яка може містити полігональну поверхню 106 і інформаційний об'єм 100. Полігональна поверхня 106 може бути представлена в пам'яті 706 множиною координат вершин, що описують множину полігонів 108a-d, чи деяким іншим способом. Інформаційний об'єм 100 може бути представлений в пам'яті 706 множиною об'ємних інформаційних точок 102a-d. Кожна з цих об'ємних інформаційних точок 102a-d може представляти собою чи включати в себе воксел.

Система 700 також містить керівний модуль 708. Керівний модуль 708 може виконувати інструкції, що виконуються комп'ютером. Керівний модуль 708 може містити інструкції, що виконуються комп'ютером 710, чи посилатися на них. Інструкції, що виконуються комп'ютером 710, можуть зберігатися в керівному модулі 708, в пам'яті 706 чи в іншому місці. Для виконання різних функцій керівний модуль 708 може виконувати інструкції, що виконуються комп'ютером 710.

Керівний модуль 708 можна конфігурувати таким чином, щоб виконувати різні функції, що виконуються комп'ютером 710. Контроль модуля 710 можна конфігурувати, щоб усунути стиснуту індексацію з полігональної поверхні 106, наприклад, шляхом окремого представлення кожної вершини кожного полігона 108a-d на полігональній поверхні 106. Керівний модуль 708 можна також конфігурувати таким чином, щоб розділити ті полігони 108a-d на полігональній поверхні 106, розмір яких перевищує максимально допустимий розмір елемента.

Керівний модуль 708 можна також конфігурувати таким чином, щоб створити текстурний елемент 300a-d для кожного полігона 108a-d, що міститься на полігональній поверхні 106. Керівний модуль 708 може зберігати текстурні елементи 300a-d в пам'яті 706 чи в іншому місці. Керівний модуль 708 можна також конфігурувати таким чином, щоб комбінувати текстурні елементи 300a-d в один чи більше текстурних атласів 400a-c. Керів-

ний модуль 708 може зберігати текстурні атласи 400a-c в пам'яті 706 чи в іншому місці.

Керівний модуль 708 можна також конфігурувати таким чином, щоб виділити сегмент 710a-c текстурного кеша 702 для кожного текстурного атласу 400a-c. Керівний модуль 708 можна також конфігурувати таким чином, щоб сканувати-конвертувати кожний полігон 108a-d на полігональній поверхні 106 на основі об'ємних інформаційних точок 102a-d, що містяться в інформаційному об'ємі 100. Результатами сканування-конвертації можуть бути один чи більше текселів, які можна записати в текселі 704 текстурного кеша 702. Текселі 704, що записані як результат сканування-конвертації певного полігона 108a, можуть бути текселями 704, що містяться в сегменті 710a, виділеному для текстурного атласу 400a, що містить схему, в яку входить полігон 108a.

Керівний модуль 708 можна також конфігурувати таким чином, щоб відображати результати сканування-конвертації для користувача за допомогою електронно-променевої трубки чи іншого екрану, їх можна роздрукувати чи відобразити іншим чином.

На Фіг.8 показаний один з прикладів здійснення способу поділу полігона і створення текстурного елемента у відповідності до здійснення даного винаходу. Цей спосіб може бути одним з прикладів здійснення кроків 206 и 208 з Фіг.2.

Цей спосіб розпочинається у кроці 800, де можна створити набір полігонів. Набір полігонів може бути представленням полігонів 108a-d, що утворюють полігональну поверхню 106. В одному з прикладів здійснення крок 800 створення набору полігонів може включати в себе усунення стиснутої індексації в представленні полігонів 108a-d. В іншому прикладі здійснення полігони можуть бути вже представлені без стиснутої індексації.

У кроці 802 можна створити набір текстурних елементів. Набір текстурних елементів можна конфігурувати таким чином, щоб він містив один чи більше текстурних елементів 300a-d. Цей набір текстурних елементів може містити максимальний розмір текстурного елемента, який можна представити шириною і висотою. При створенні набір текстурних елементів може не містити жодних текстурних елементів 300a-d.

У кроці 804 досліджується набір полігонів, щоб визначити, чи не залишилися в наборі полігонів необроблені полігони 108a-d. Якщо набір полігонів не містить жодних необроблених полігонів, то спосіб завершений, і закінчується він у кроці 806.

Якщо набір полігонів містить один чи більше необроблених полігонів 108a-d, то у кроці 808 вибирається один з необроблених полігонів 108a-d. У кроці 810 полігон 108a-d досліджується, щоб визначити, чи узгоджується полігон 108a-d з текстурним елементом 300a-d максимально допустимого розміру. Якщо полігон 108a-d не узгоджується з текстурним елементом 300a-d максимально допустимого розміру, то полігон 108a-d може бути розділений у кроці 812. Розділення може здійснюватися за допомогою будь-якого процесу, в якому утворюється кілька менших полігонів 108a-d, об'єднанням яких є оригінальний полігон 108a-d. В

одному з прикладів здійснення полігон 108a-d ділиться навпіл по довшій грані полігона 108a-d і утворюється нова грань у точці поділу вершини полігона 108a-d. В іншому прикладі здійснення полігон 108a-d можна розділити шляхом вирождення центра ваги. Це може включати в себе знаходження центра ваги полігона 108a-d і утворення нової грані з кожної вершини полігона 108a-d до центра ваги.

У кроці 814 полігони 108a-d, одержані з поділу, можна помістити в набір полігонів. Це може забезпечити поділ кожного з цих полігонів 108a-d, у випадку необхідності, і створення для кожного з них текстурного елемента 300a-d.

У кроці 816 оригінальні полігони 108a-d можна позначити як оброблені. Це може включати в себе, наприклад, встановлення логічної змінної чи позначення іншим способом, що полігон 108a-d вже досліджений. Не потрібно створювати текстурний елемент 300a-d створювати для оригінального полігона 108a-d, оскільки будуть створені текстурні елементи 300a-d для полігонів 108a-d, одержаних з поділу.

Якщо у кроці 810 було визначено, що полігон 108a-d, що досліджується, узгоджується з текстурним елементом 300a-d максимально допустимого розміру, то у кроці 818 можна створити новий текстурний елемент 300a-d. Цей створений текстурний елемент 300a-d може включати в себе полігон 108a-d чи може містити інформацію, що описує полігон 108a-d. В одному з прикладів здійснення можна вибрати розміри текстурного елемента 300a-d, щоб скоротити розмір текстурного елемента 300a-d, забезпечивши, щоб текстурний елемент 300a-d містив полігон 108a-d. Наприклад, В одному з прикладів здійснення, текстурним елементом 300a-d може бути найменший прямокутний текстурний елемент 300a-d, що містить полігон 108a-d.

У кроці 820 текстурний елемент 300a-d можна додати в набір текстурних елементів. Потім спосіб продовжується у кроці 816, де полігон 108a-d позначається як оброблений. Спосіб повертається до кроку 804, де визначається, чи не залишилися в наборі полігонів якісь необроблені полігони 108a-d.

На Фіг.9 показана блок-схема, що представляє один з прикладів здійснення способу створення текстурного атласу у відповідності до кроку 210 з Фіг.2. Цей спосіб розпочинається з кроку 900, де можна створити набір текстурних атласів. Набір текстурних атласів можна конфігурувати таким чином, щоб він містив один чи більше текстурних атласів 400a-c. Набір текстурних атласів може містити максимальний розмір атласу, який можна представити шириною і висотою. При створенні набір текстурних атласів може не містити жодних текстурних атласів 400a-c.

У кроці 902 можна дослідити набір текстурних елементів, щоб визначити, чи не залишилися в наборі текстурних елементів необроблені текстурні елементи 300a-d. Досліджений набір текстурних елементів може представляти собою набір текстурних елементів, створений за допомогою способу з Фіг.8. Якщо набір текстурних елементів не містить жодних необроблених текстурних елементів

300a-d, то спосіб завершений, і закінчується він у кроці 904.

Якщо набір полігонів містить один чи більше необроблених текстурних елементів 300a-d, то у кроці 906 вибирається один з необроблених текстурних елементів 300a-d. У кроці 908 текстурний елемент 300a-d і набір атласів досліджуються, щоб визначити, чи узгоджується текстурний елемент 300a-d з текстурним атласом 400a-c, що міститься в наборі текстурних атласів. Якщо текстурний елемент 300a-d не узгоджується з жодним текстурним атласом 400a-c, то у кроці 910 можна створити новий текстурний атлас 400a-c. Цей новий текстурний атлас 400a-c можна помістити в набір текстурних атласів. В одному з прикладів здійснення порожнє місце в текстурному атласі 400a-c представляється порожніми текстурними елементами 300a-d у текстурному атласі 300a-c. Це можуть бути текстурні елементи 300a-d у, що конкретно позначені як порожні. У цьому прикладі здійснення у текстурний атлас 400a-c можна додати один порожній текстурний елемент 300a-d з розмірами текстурного атласу 400a-c.

У кроці 912 текстурний елемент 300a-d вибраний у кроці 906, можна додати в текстурний атлас 400a-c в наборі текстурних атласів. У одному з прикладів здійснення розміщення текстурного елемента 300a-d можна виконувати оптимальним способом чи іншим чином, щоб скоротити порожнє місце в текстурних атласах 400a-c.

У одному з прикладів здійснення порожнє місце в кожному текстурному атласі 400a-c представляється порожніми текстурними елементами 300a-d. У цьому прикладі здійснення текстурний елемент 300a-d, що додається в набір текстурних атласів, можна помістити в положення порожнього текстурного елемента 300a-d, що має найменшу різницю площі з доданим текстурним елементом 300a-d. Після додавання текстурного елемента

300a-d можна створити нові порожні текстурні елементи 300a-d. Ці нові порожні текстурні елементи 300a-d можуть покривати площу текстурного атласу 400a-c, що був попередньо покритий порожнім текстурним елементом 300a-d, проте не є покритий текстурним елементом 300a-d, що був доданий. Ці нові порожні текстурні елементи 300a-d можна помістити в положення різниці площі між доданим текстурним елементом 300a-d і порожнім текстурним елементом 300a-d.

У кроці 914 текстурний елемент 300a-d можна позначити як оброблений. Це може включати в себе, наприклад, встановлення логічної змінної чи позначення іншим способом, що текстурний елемент 300a-d вже доданий у набір текстурних атласів. Спосіб повертається до кроку 902, де визначається, чи не залишились в наборі текстурних елементів якісь необроблені текстурні елементи 300a-d.

Подальший опис даного винаходу є ілюстративним, і особи, що мають знання у цій області техніки, можуть вносити зміни у конфігурацію і здійснення. Наприклад, можна комбінувати кроки чи виконувати їх в будь-якій послідовності. Незважаючи на те що текстурні елементи і текстурні атласи описані як фізичні сегменти електронної пам'яті, здатні містити один чи більше полігонів, вони необов'язково повинні фізично містити цю пам'ять. Швидше, текстурні елементи чи текстурні атласи можуть містити дані, що визначають розміри достатнього сегмента пам'яті. Апаратне забезпечення, програмне забезпечення чи інші ресурси, описані як сингулярні, у варіантах здійснення можуть бути розподіленими; подібним чином, ресурси, описані як розподілені, у варіантах здійснення можуть бути комбінованими. Обсяг даного винаходу відповідним чином обмежується лише наступною формулою.

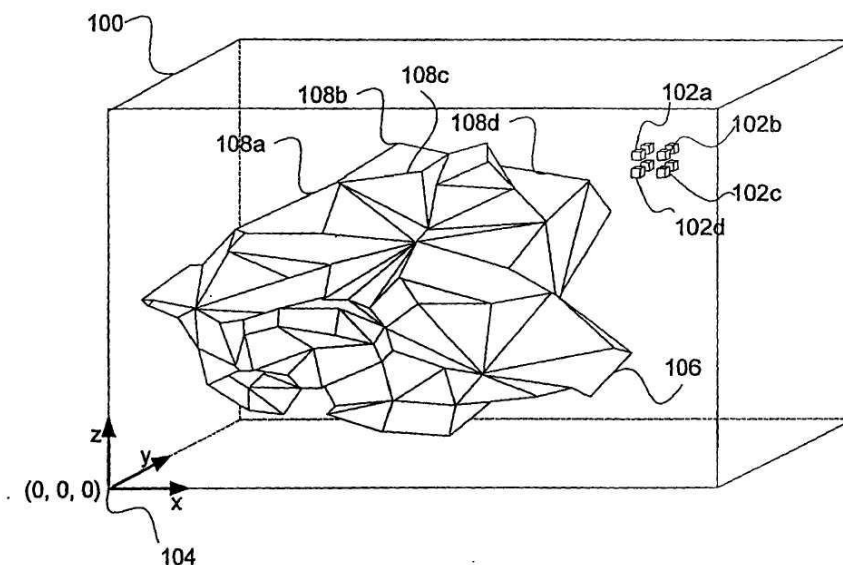
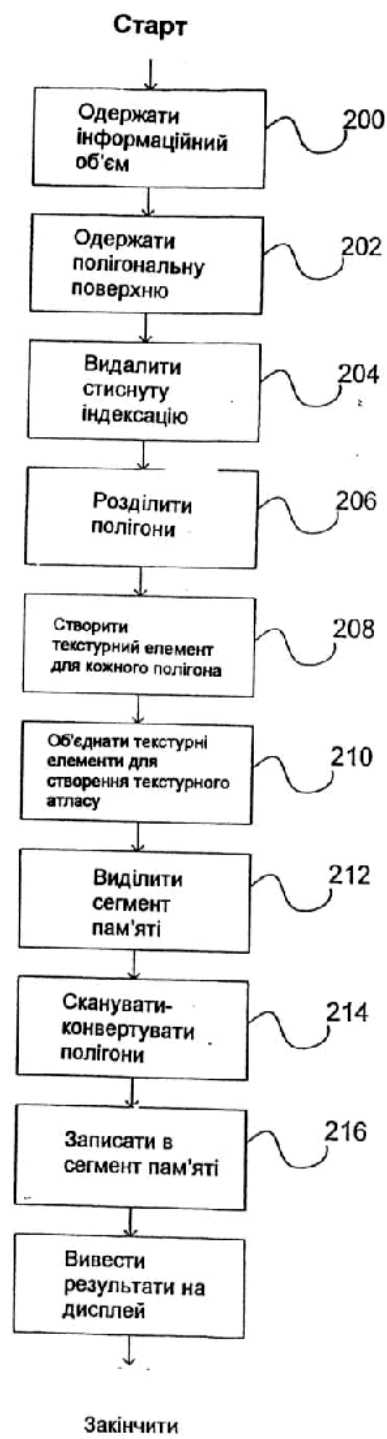


Fig. 1



Фіг. 2

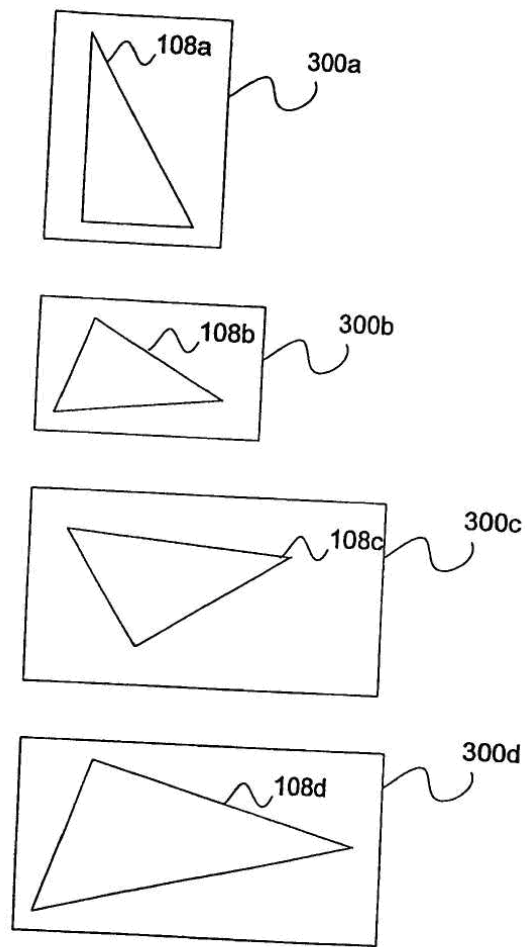


Fig. 3

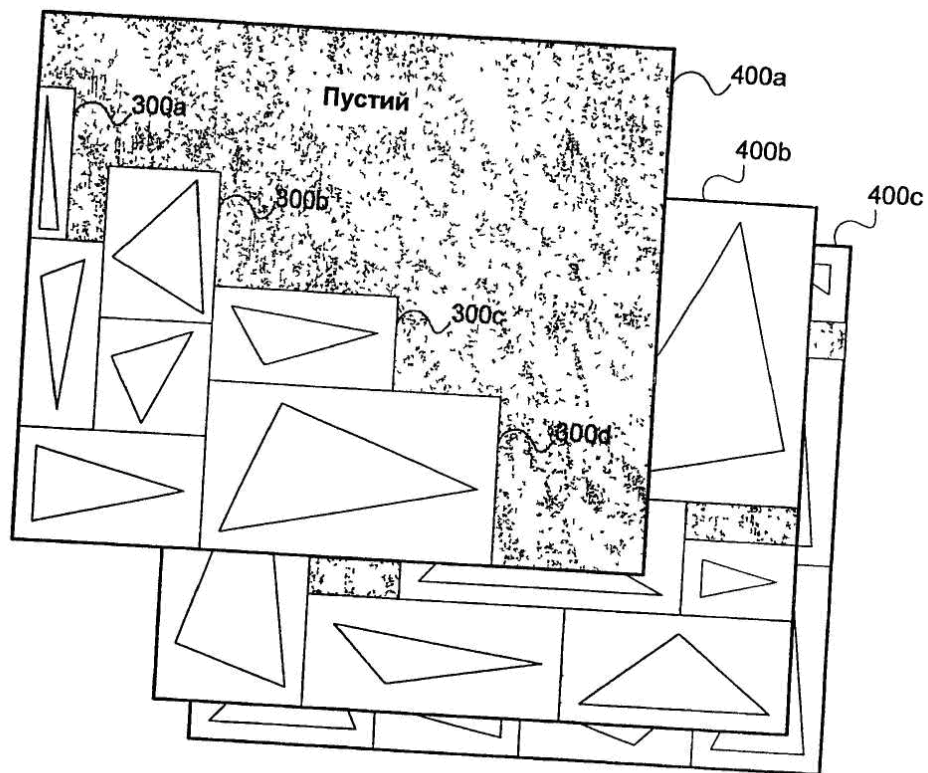


Fig. 4

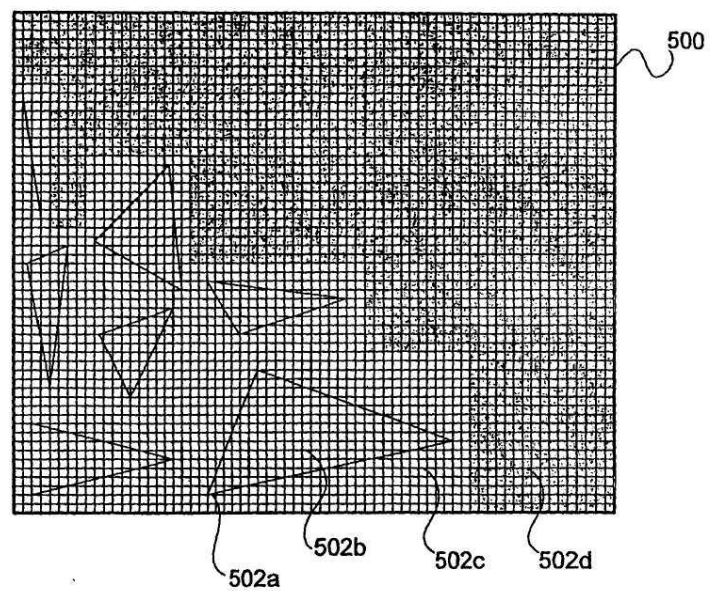


Fig. 5

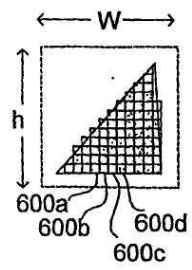
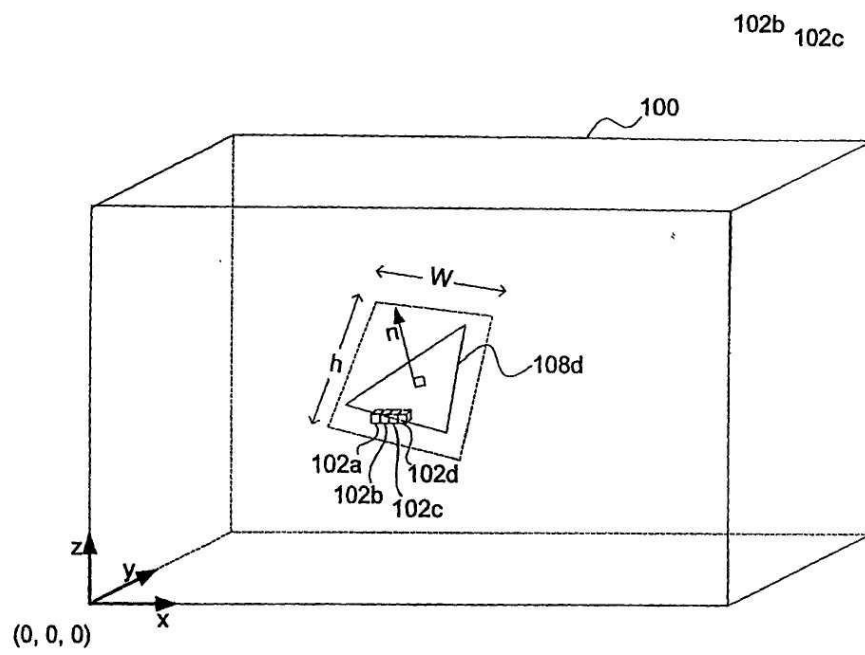
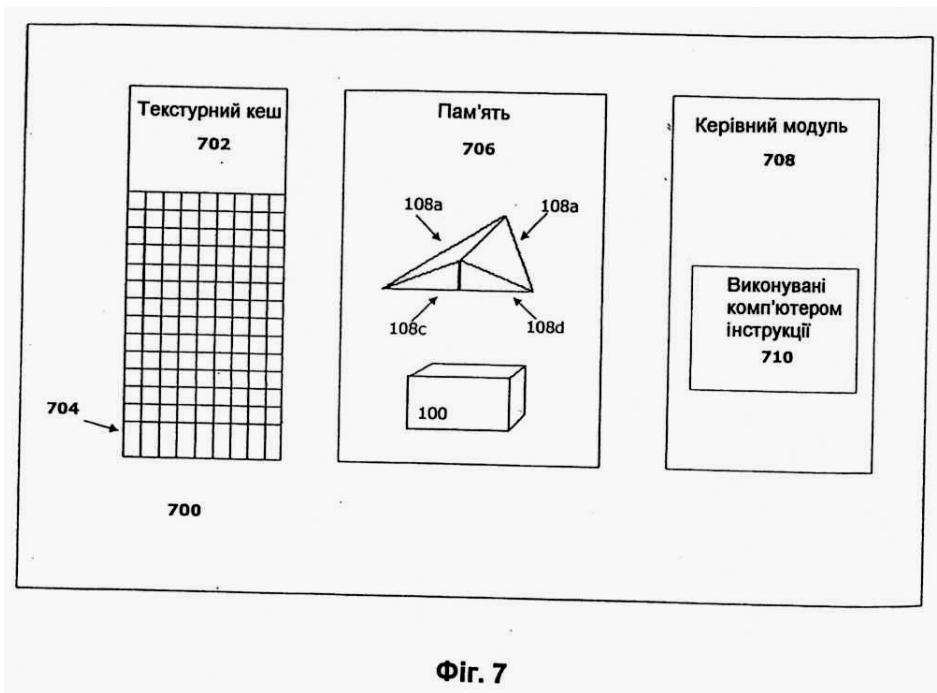
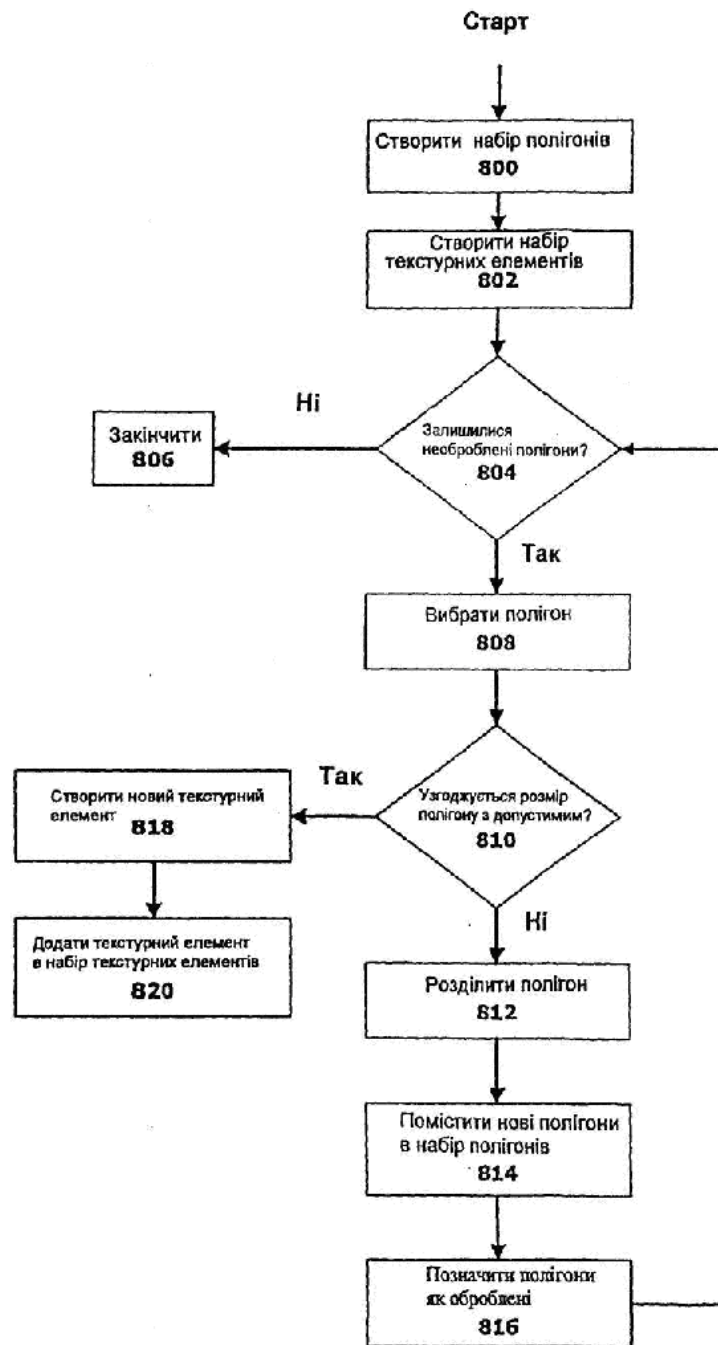
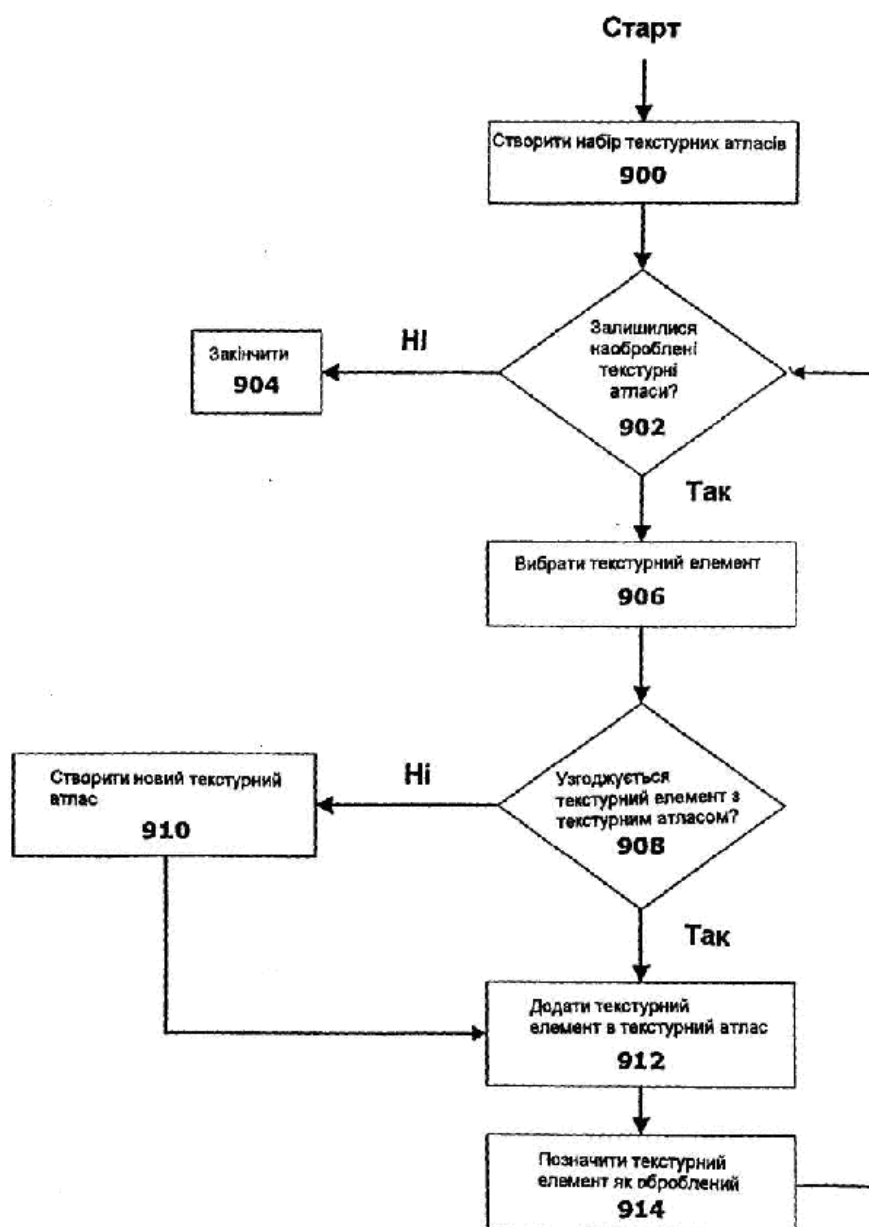


Fig. 6





Фіг. 8



Фіг. 9