

УДК 004.925

В. М. ГУСЯТИН, М. В. ГУСЯТИН

Харьковский национальный университет радиоэлектроники, Украина

## СИНТЕЗ ОБРАТНЫМ ТРАССИРОВАНИЕМ ИЗОБРАЖЕНИЙ ВЕКТОРНЫХ ТЕКСТУР, СФОРМИРОВАННЫХ МЕТОДОМ СФЕРИЧЕСКОЙ ИНТЕРПОЛЯЦИИ

*В статье рассмотрены вопросы повышения реалистичности сцен, синтезируемых методом обратного трассирования. Решается задача визуализации обратным трассированием векторных текстур, заданных прямыми и сглаженных методом сферической интерполяции. Приведены аналитические соотношения для реализации каждого этапа визуализации таких векторных текстур. Особенностью решения задачи является совмещение метода сферической интерполяции и синтеза изображения векторной текстуры обратным трассированием за один проход. Такой подход позволяет расширить области совместного применения этих методов, в том числе, на основе использования существующих баз данных векторных текстур, заданных прямыми линиями.*

**Ключевые слова:** метод обратного трассирования, моделирование кривых и поверхностей, квадрик, интерполяция, текстура.

### Введение

В современной компьютерной графике главным направлением является повышение реалистичности синтезируемых сцен [1, 3]. Выполнение требования высокой реалистичности является важным фактором, например, при проектировании систем визуализации тренажеров транспортных средств различного назначения (авиатренажеры и др.), при создании художественных фильмов с применением компьютерной графики и др. Одним из возможных путей решения этой проблемы является применение метода обратного трассирования при решении задач синтеза изображений [2, 3].

Метод обратного трассирования активно развивается в последние годы [3]. Этот метод позволяет синтезировать изображение аналитически описываемых поверхностей без их предварительной триангуляции, что существенно повышает реалистичность синтезируемых изображений объектов. В тоже время накопленная в компьютерной графике база моделей объектов и программного обеспечения, ориентирована на линейно заданные кривые и триангулированные поверхности, без которых не работает классический метод прямого трассирования (метод растеризации).

Возникает задача совместимости метода обратного трассирования с накопленной базой моделей и программного обеспечения с сохранением его лучших особенностей.

Решение этой задачи существенно упрощается, если для интерполяции исходной триангулированной поверхности используются алгебраические кривые или поверхности, записанные в неявной форме [10].

Таким образом, решение задачи совмещения новых методов моделирования гладких геометрических форм плоских кривых, заданных прямыми линиями, и метода обратного трассирования является актуальной задачей на данном этапе развития компьютерной графики.

### 1. Анализ литературы

Реалистичность синтезируемой сцены во многом определяется методом моделирования кривых и криволинейных поверхностей, интерполирующих точки, которые заданы на плоскости или в пространстве. Различными авторами разработан ряд методов решения этой задачи [1-8, 10]. Анализ параметрического описания, предусматривает, что исходной позицией проекционного луча, строящего изображение, является точка на кривой или поверхности [4, 5]. Это затрудняет применение стандартных алгоритмов синтеза изображений с использованием метода обратного трассирования. Такая особенность параметрического описания заставила исследователей обратиться к алгебраическим кривым и поверхностям [6-8, 10]. Авторами работ [6, 7] предлагается использовать для интерполяции алгебраические поверхности высоких порядков. Недостатком такого подхода является сложность расчета коэффициентов таких поверхностей. В [8] предлагается для интерполяции исходной сетки высокой детализации использовать алгебраическую поверхность второго порядка. По мнению автора этого метода, он может быть использован в системах интерактивного объемно ориентированного геометрического моделирования. В работе [10] предлагается использовать метод сферической ин-

терполяції, який оснований на використанні простейшого квадрата – окружности либо сферы. При таком подходе возможно решение следующих задач. При использовании существующей базы моделей с триангулированной поверхностью возможно восстановление гладкой поверхности. В этом случае отсутствует необходимость в интерполяции освещенности по Гуро (Gouraud shading) или Фонгу (Phong shading). При расчете бликов нет необходимости в расчете BRDF (Bidirectional reflectance distribution function), а также при расчете освещенности учитывается перспектива. При отображении имеем неискаженную граничную форму теней. Упрощается нанесение текстуры и многое другое.

Такой подход упрощает нахождение точки пересечения проекционного луча с интерполирующей поверхностью методом обратного трассирования.

## 2. Цель статьи

Излагается метод синтеза изображений обратным трассированием векторных текстур, сформированных методом сферической интерполяции.

## 3. Этапы синтеза изображения

1. *Классификационное описание.* Для уменьшения времени счета методом обратного трассирования, наряду с обычным описанием объектов (геометрия, цвет и др.), целесообразно вводить классификационное описание [9].

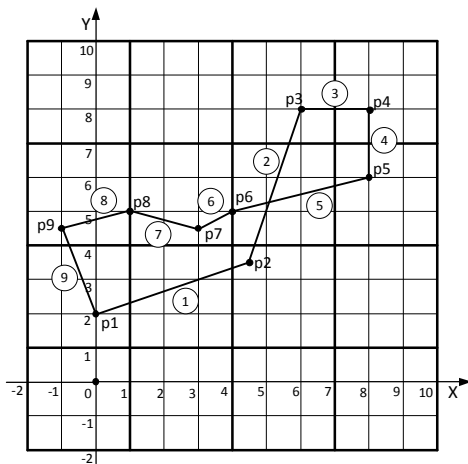


Рис. 1. Приведен пример заготовки плоской кривой в виде замкнутой ломаной линии (точки  $p_1 - p_9$ )

Линия на рис. 1 для упрощения примера охвачена квадратом, который разбит на квадраты меньшего размера (домены). Каждому домену в качестве идентификатора присваивается, например, значение координат левой нижней точки (см. список 1). В списках list1 и list2, приведенных ниже исключена (для сокращения) координата  $z = 0$

$$\text{list1} = \left\{ \begin{array}{l} \{-2, -2\}, \{1, -2\}, \{4, -2\}, \{7, -2\}, \{-2, 1\}, \\ \{1, 1\}, \{4, 1\}, \{7, 1\}, \{-2, 4\}, \{1, 4\}, \{4, 4\}, \\ \{7, 4\}, \{-2, 7\}, \{1, 7\}, \{4, 7\}, \{7, 7\} \end{array} \right\}.$$

Суть классификационного описания [9] заключается в составлении списка2, в котором с каждым доменом связываются отрезки ломаной линии, пересекающие данный домен.

$$\text{list2} = \left\{ \begin{array}{l} \{-2, 2\}, \{ \}, \{1, -2\}, \{ \}, \{4, -2\}, \{ \}, \{7, -2\}, \{ \}, \\ \{-2, 1\}, \{pv[1], pv[9]\}, \{1, 1\}, \{pv[1]\}, \\ \{4, 1\}, \{pv[1], pv[2]\}, \{7, 1\}, \{ \}, \{-2, 4\}, \{pv[9], \\ pv[8]\}, \{1, 4\}, \{pv[7], pv[6]\}, \{4, 4\}, \{pv[5], pv[2]\}, \\ \{7, 4\}, \{pv[5], pv[4]\}, \{-2, 7\}, \{ \}, \{1, 7\}, \{ \}, \\ \{4, 7\}, \{pv[2], pv[3]\}, \{7, 7\}, \{pv[3], pv[4]\}, \end{array} \right\}.$$

В этом списке каждый отрезок  $pv$  пронумерован, например, в соответствии с рисунком 1. В свою очередь каждый отрезок задан в виде списка, в который входят координаты точек конца этого отрезка и вектора нормалей в этих точках для синтезируемой интерполирующей кривой [10]. Ниже в качестве примера приведена запись для отрезка 1 (номера отрезков обозначены в кружках на рисунке 1)

$$pv1 = \{ \{p1, v1\}, \{p2, v2\} \}. \quad (1)$$

При отсутствии классификационного описания, приходится использовать силовой алгоритм, при котором каждый проекционный луч необходимо тестировать со всеми отрезками векторной текстуры.

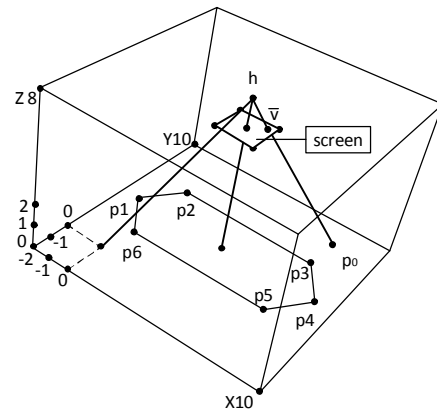


Рис. 2. Геометрические элементы задачи

2. *Область интерполяции.* В соответствии с [10] метод сферической интерполяции позволяет построить плоскую гладкую кривую, как локально для отдельного отрезка, так и в целом для всех отрезков. Поэтому в дальнейшем, излагая основные этапы синтеза, мы рассматриваем процесс вычислений для локального (отдельного) отрезка.

На рис. 2 показана правая декартова система координат с/к XYZ. Геометрическая схема реализу-

ет центропроективное преобразование синтезируемой векторной текстуры на экран. Далее полагаем, что все геометрические элементы схемы наблюдателя переведены из с/к наблюдателя в базовую с/к XYZ, в которой задана заготовка векторной текстуры, например на плоскости  $z = 0$ . Для упрощения дальнейшего изложения (без потери общности решения) зафиксировано положение наблюдателя таким образом, что экран расположен параллельно карте текстуры. Поскольку в дальнейшем в соотношениях ниже будут записаны только векторные величины - не будем обозначать их стрелкой.

На рисунке приняты обозначения:

$h$  - радиус- вектор центра проекции;

$v$  - вектор наблюдения, соединяет точку  $h$  с произвольным пикселем экрана;

$p_0$  - точка пересечения проекционного луча с поверхностью  $z = 0$ .

По аналогии с [10] выделим для каждого отрезка область интерполяции с помощью плоскостей проведенных через точки, например,  $p_1, p_2$  перпендикулярно этому отрезку. Запишем функции для этих плоскостей в векторной форме в виде скалярного произведения.

$$\begin{aligned} f1(r) &= (r - p1)(p2 - p1), \\ f2(r) &= (r - p2)(p2 - p1), \end{aligned} \quad (2)$$

где  $r$  - произвольный радиус-вектор.

Область интерполяции для отрезка находится в пространстве между плоскостями (2) и позволяет выбрать из общего числа отрезков, попавших в домен, с наибольшей вероятностью те отрезки, которые следует тестировать.

3. Основные соотношения для выбора домена и области интерполяции. Запишем в параметрической форме уравнение прямой (в дальнейшем проекционный луч), совпадающей с вектором наблюдения.

$$p(t) = h + v * t, \quad (3)$$

где  $t$  - параметр-скаляр.

Найдем точку пересечения проекционного луча с поверхностью карты текстуры (это может быть карта достаточно большого размера, например, разметка площади аэропорта: ВПП, рулежных дорожек, стоянок самолетов и др.).

В нашем случае соотношение для этой поверхности имеет вид

$$r * n = 0, \quad (4)$$

где  $n = \{0,0,1\}$  - единичный вектор нормали к поверхности (4).

Найдем совместное решение уравнений (3) и (4) относительно  $t$  и подставим полученное значение в (3). Таким образом, получим радиус- вектор  $p(t_0)$ , который определяет при заданном значении

$t_0$  положение точки пересечения  $p_0(x, y, z)$  проекционного луча с поверхностью (4) - в дальнейшем  $p_0$ .

Для определения домена, в котором оказалась точка  $p_0$ , запишем компоненты вектора  $p(t_0)$  через проекции на оси с/к XYZ.

$$p_0 = \{h_x + v_x t_0, h_y + v_y t_0, h_z + v_z t_0, \}. \quad (5)$$

По выделенной целой части компонент определяем из списка 1 домен, в котором разместились точка  $p_0$ . Далее из классификационного описания (список 2) определяем отрезки  $pv$  заготовки векторной текстуры, которые пересекли или оказались в данном домене (рисунок 1).

Для выбранных отрезков определяем те области интерполяции, в которые попала точка  $p_0$ .

В соотношения (2) подставляем значение радиус- вектора для середины отрезка

$$p12 = (p1 + p2) / 2. \quad (6)$$

По результатам этих подстановок формируем набор «знаков-эталонов» для каждой из областей

$$f_e = \text{sign}\{f1(p12), f2(p12)\}. \quad (7)$$

Затем в соотношения (2) подставляется  $p_0$  и тогда имеем

$$f_0 = \text{sign}\{f1(p_0), f2(p_0)\}. \quad (8)$$

Используя соотношения (8) и набор «знаков-эталонов» (7), можно для любой точки  $p_0$  на плоскости карты векторной текстуры идентифицировать (указать) ее положение по отношению к областям интерполяции путем сравнения ее знаков со «знаками-эталонами». Для выполнения оценки этих операций будем вычислять предикат

$$pr1 = \begin{cases} 1, & (f_e = f_0) - \text{счет}, \\ 0, & \text{иначе} - \text{цвет}_\text{фона}. \end{cases} \quad (9)$$

Если  $pr1 = 0$ , то пикселю присваивается цвет фона, в противном случае счет по данному проекционному лучу продолжается.

Такой подход позволяет уменьшить число выбранных отрезков в домене, в ряде случаев, до одного, а также прекратить счет для данного проекционного луча и присвоить цвет фона соответствующему пикселю, так как существует много фонового пространства между областями интерполяции. В целом это уменьшает время счета. Для уменьшения счета в дальнейшем целесообразно сделать оценку расстояния между точкой  $p_0$  и точками  $p_1, p_2$  концов отрезка. Введем предикат

$$pr2 = \begin{cases} 1, & (|p1 - p_0| \text{ or } |p2 - p_0|) \leq w_0, \\ 0, & \text{иначе} - \text{счет}, \end{cases} \quad (10)$$

где  $w_0$  - некоторое значение, определяющее ширину синтезируемой кривой.

В случае, если  $rg_2 = 1$ , пикселю задается цвет кривой, иначе - счет по данному проекционному лучу продолжается.

4. Основные соотношения для построения кривой в процессе ее отображения методом обратного трассирования. Для упрощения обозначений будем считать, что на предыдущем этапе выбран отрезок  $p_1, p_2$ .

Векторное поле направляющих, на которых собственно и будут лежать точки интерполирующей кривой, построенной методом сферической интерполяции, определим, как прямые, расположенные перпендикулярно к прямой линии, соединяющей две последовательно заданные точки  $p_1, p_2$  в области интерполяции [10]. Полные данные для отрезка  $l$  определены в  $pv_1$ .

Основной особенностью метода сферической интерполяции является использование такого квадрата как окружность либо сфера для построения кривой произвольной формы между двумя точками. Наличие нормалей к будущей кривой в каждой точке, через которые она проходит, позволяет получить наряду со связностью (C0) и непрерывность первой производной т.е. гладкость (C1) синтезируемой кривой [1, 10]. В процессе синтеза изображения кривой выполним следующие геометрические построения. Через концы отрезка каждой прямой, например, через точки  $p_1, p_2$  проведем две сферы (так называемые опорные сферы). При этом центр  $o_1$  одной сферы должен находиться на прямой, проходящей через точку  $p_1$  и совпадающей с вектором  $v_1$  нормали в этой точке. Центр  $o_2$  второй сферы должен находиться на прямой, проходящей через точку  $p_2$  и совпадающей с вектором нормали  $v_2$ .

Соотношения для уравнений этих прямых имеют вид

$$\begin{aligned} p_{o1} &= p_1 + v_1 * t_1, \\ p_{o2} &= p_2 + v_2 * t_2. \end{aligned} \quad (11)$$

Запишем уравнение для плоскости, проходящей через точку (6) перпендикулярно этому отрезку

$$f_{12}(r) = (r - p_{12})(p_2 - p_1) = 0. \quad (12)$$

Найдем совместное решение уравнений (11) и (12) относительно  $t_1, t_2$  и подставим полученные значения в (11). Таким образом, получим радиус-вектора для центров опорных сфер  $o_1, o_2$ .

Запишем скалярные значения радиусов для опорных сфер

$$R_1 = |p_1 - o_1|, R_2 = |p_2 - o_2|. \quad (13)$$

В окончательном виде запишем уравнения для опорных сфер в векторной форме

$$fs_1(r) = |r - o_1| - R_1 = 0, \quad (14)$$

$$fs_2(r) = |r - o_2| - R_2 = 0.$$

Построим векторное поле направляющих [10]. Для этого запишем уравнение плоскости, проходящей через точку  $p_0$  перпендикулярно вектору отрезка  $p_2 - p_1$

$$(r - p_0)(p_2 - p_1) = 0. \quad (15)$$

Запишем уравнение прямой параллельной вектору отрезка  $p_2 - p_1$

$$p(t) = p_1 + (p_2 - p_1)t. \quad (16)$$

Найдем совместное решение уравнений (15) и (16) относительно  $t$  и подставим полученное значение  $\hat{t}$  в (16). Таким образом, получим радиус-вектор  $p(\hat{t})$  для точки пересечения.

Запишем уравнение прямой, проходящей через точки  $p(\hat{t})$  и  $p_0$

$$p(t) = p(\hat{t}) + (p_0 - p(\hat{t}))t. \quad (17)$$

Уравнение (17) является направляющей прямой, на которой и находится точка синтезируемой кривой.

Найдем совместное решение уравнения (17) со сферами (14) относительно  $t$ . Здесь следует отметить, что для каждой сферы имеется две точки пересечения. Обозначим параметр этих точек.

Пусть это будут: для первой окружности  $t_{11}, t_{12}$  и для второй  $t_{21}, t_{22}$ . Подставим эти значения в уравнение (17) и получим радиус-вектора точек пересечения для первой и второй окружности соответственно

$$p(t_{11}), p(t_{12}), p(t_{21}), p(t_{22}). \quad (18)$$

Для правильного выбора корней в решениях (18) запишем соотношение для вектора

$$v_{o1} = o_1 - p_{12}. \quad (19)$$

Определим функцию, используя уравнение плоскости, проведенной через точку  $p_{12}$  перпендикулярно вектору  $v_{o1}$

$$p_l(r) = (r - p_{12})v_{o1}. \quad (20)$$

Подставим в равенство (20) координаты центров окружностей  $o_1$  и  $o_2$ . По результатам этих подстановок получим эталонные значения знаков:

$$\begin{aligned} s_{o1} &= \text{sign}(p_l(o_1)), \\ s_{o2} &= \text{sign}(p_l(o_2)). \end{aligned} \quad (21)$$

Подставляя (18) в (20) и попарно сравнивая результат с (21), выбираем из пары корней тот, который имеет знак, противоположный знакам  $s_{o1}$  и  $s_{o2}$ . Пусть, например, это будут  $p(t_{11})$  и  $p(t_{21})$ .

С учетом (16) запишем выражения для векторов

$$s_{11} = p(t_{11}) - p(\hat{t}), s_{21} = p(t_{21}) - p(\hat{t}). \quad (22)$$

Окончательно, общее выражение для радиуса - вектора, задающего точку на синтезируемой интер-

полирующей кривой, например, для отрезка между точками  $p_1, p_2$ , запишем в виде соотношения:

$$r_{12}(t) = p(t) + s_{11}(1-t) + s_{21}t, \quad (23)$$

где  $t$  - параметр, изменяющейся от нуля до единицы. Шаг этого параметра определяется шагом  $\hat{t}$ , для которого этот шаг зависит от разрешения экрана. Полученное соотношение соответствует работе авторов в [10].

Для задания ширины синтезируемой кривой определим модуль вектора

$$w = |r_{12} - p_0|. \quad (24)$$

Введем предикат

$$pr3 = \begin{cases} 1, & (w \leq w_0), \\ 0, & \text{иначе.} \end{cases}$$

В случае, если  $pr3 = 1$ , пикселю задается цвет кривой, иначе - цвет фона.

Окончательно цвет пикселя определяется переключатальной функцией

$$cp = pr1 \cap (pr2 \cup pr3). \quad (25)$$

В случае, если  $cp = 1$ , пикселю задается цвет кривой, иначе - цвет фона.

5. Математическое моделирование. Вопросы погрешности интерполяции рассмотрены авторами ранее в [10].

На рисунках 3 и 4 представлены примеры моделирования в пакете Wolfram Mathematica задачи синтеза обратным трассированием изображений плоской кривой, сформированной методом сферической интерполяции.

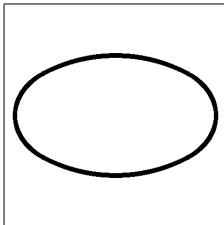


Рис. 3. Визуализация эллипса. Исходные данные: точки эллипса  $p_1 - p_6$  отображены на рис. 2

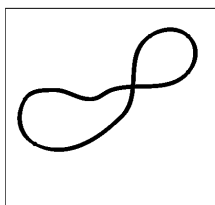


Рис. 4. Визуализация интерполирующей кривой, в соответствии с методом. Исходные данные: произвольные точки  $p_1 - p_9$  (рис. 1)

### Выводы

Особенностью решения является совмещение метода сферической интерполяции и синтеза изображения обратным трассированием за один про-

ход. Получены аналитические соотношения для реализации каждого этапа. Приведены результаты моделирования в пакете Wolfram Mathematica, подтверждающие правильность принятых решений.

Дальнейшая работа авторов направлена на разработку визуализации этого типа пространственных кривых и поверхностей методом обратного трассирования. Такой подход позволяет расширить области применения метода обратного трассирования в компьютерной графике с сохранением его лучших особенностей в сравнении с классическим методом синтеза изображений.

### Литература

1. *Computer Graphics (principles and practice) [Text] / John F. Hughes, Andries van Dam, Morgan McGuire, David F. Sklar, James D. Foley, Steven K. Feiner, Kurt Akeley. – Addison-Wesley Publishing Company, Inc., 2014. – 1209 p.*
2. Никулин, Е. А. Компьютерная геометрия и алгоритмы машинной графики [Text] / Е. А. Никулин. – Санкт-Петербург : «БХВ-Петербург», 2003. – 550 с.
3. Hurley, Jim. Ray Tracing Goes Mainstream [Text] / Jim Hurley // Intel Technology Journal “Compute-Intensive, Highly Parallel Applications and Uses” – 2005. – Vol. 9, Iss. 2. – P. 99–107.
4. Barth, W. Efficient Ray Tracing for Bezier and B-Spline Surfaces [Text] / W. Barth and W. Sturzlinger // Computers & Graphics. – 1993. – No. 17(4). – P. 423–430.
5. Campanga, S. Improving Bezier Clipping and Chebyshev Boxing for Ray Tracing Parametric Surfaces [Electronic resource] / S. Campanga, P. Slusallek // Technical report, University of Erlangen, Computer Graphics Group, 1996. – Access mode: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.30.1448&rep=rep1&type=pdf>. – 12.05.2015
6. Bajaj, C. Higher-order interpolation and least-squares approximation using implicit algebraic surfaces [Text] / C. Bajaj, I. Ihm, J. Warren // ACM Transactions on Graphics. – 1993. – Vol. 12, Iss. 4. – P. 327–347.
7. Bajaj, C. Modeling with Cubic A-Patches [Text] / C. Bajaj, J. Chen, G. Xu // ACM Transactions on Graphics. – 1995. – Vol. 14, Iss. 2. – P. 103–133.
8. Вяткин, С. И. Моделирование сложных поверхностей с применением функций возмущения [Текст] / С. И. Вяткин // Автометрия. – 2007. – № 3. – С. 40–47.
9. Гусятин, В. М. Классификация теневого пространства и синтез теней, формируемых рельефом [Текст] / В. М. Гусятин, А. Н. Бугрий // Вестник НТУ “ХПИ”. – 2003. – № 26. – С. 53–57.
10. Гусятин, М. В. Построение плоской кривой методом сферической интерполяции в задачах компьютерной графики [Текст] / М. В. Гусятин, В. М. Гусятин // Системи обробки інформації. – 2013. – № 1(108). – С. 160–163.

### References

1. Hughes, John F., Morgan McGuire, Andries van Dam, Sklar, David F., Foley, James D., Feiner, Steven K., Akeley, Kurt. *Computer Graphics (principles and practice)*. Addison-Wesley Publishing Company, Inc., 2014. 1209 p.
2. Nikulin, E. A. *Komp'yuternaya geometriya i algoritmy mashinnoi grafiki* [Computer geometry and algorithms of computer graphics]. St. Petersburg, "BHV-Petersburg" Publ., 2003. 550 p.
3. Hurley, Jim. Ray Tracing Goes Mainstream. *Intel Technology Journal "Compute-Intensive, Highly Parallel Applications and Uses"*, 2005, vol. 9, iss. 2, pp. 99–107.
4. Barth, W., Sturzlinger, W. Efficient Ray Tracing for Bezier and B-Spline Surfaces, *Computers & Graphics*, 1993, no. 17(4), pp. 423–430.
5. Campanga, S., Slusallek P. Improving Bezier Clipping and Chebyshev Boxing for Ray Tracing Parametric Surfaces. *Technical report, University of Erlangen, Computer Graphics Group*, 1996. Available at: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.30.1448&rep=rep1&type=pdf> (accessed 22.05.2012)
6. Bajaj, C., Ihm, I., Warren, J. Higher-order interpolation and least-squares approximation using implicit algebraic surfaces. *ACM Transactions on Graphics*, 1993, vol. 12, iss. 4, pp. 327–347.
7. Bajaj, C., Chen, J., Xu, G. Modeling with Cubic A-Patches. *ACM Transactions on Graphics*, 1995, vol. 14, iss. 2, pp. 103-133.
8. Vyatkin, S. I. Modelirovanie slozhnykh poverkhnostei s primeneniem funktsii vozmushcheniya [Modelling of complex surfaces using perturbation functions]. *Avtometriya*, 2007, no. 3, pp. 40–47.
9. Gusyatin, V. M., Bugrii, A. N. Klassifikatsiya tenevogo prostranstva i sintez tenei, formiruemykh rel'efom [Classification shadow space and synthesis of shadows formed by relief]. *Vestnik NTU "KhPI"*, 2003, no. 26, pp. 53-57.
10. Gusyatin, M. V., Gusyatin, V. M. Postroenie ploskoi krivoi metodom sfericheskoi interpolyatsii v zadachakh komp'yuternoi grafiki [Construction of spherical plane curve interpolation method in computer graphics problems]. *Sistemi obrobki informacii – Information processing systems*, 2013, no. 1(108), pp. 160-163.

Поступила в редакцію 6.02.2016, рассмотрена на редколлегии 18.02.2016

### СИНТЕЗ ЗВОРОТНИМ ТРАСУВАННЯМ ЗОБРАЖЕНЬ ВЕКТОРНИХ ТЕКСТУР, СФОРМОВАНИХ МЕТОДОМ СФЕРИЧНОЇ ІНТЕРПОЛЯЦІЇ

*В. М. Гусятін, М. В. Гусятін*

У статті розглянуто питання підвищення реалістичності сцен, синтезованих методом зворотного трасування. Вирішується задача візуалізації зворотним трасуванням векторних текстур, заданих прямими і згладжених методом сферичної інтерполяції. Наведено аналітичні співвідношення для реалізації кожного етапу візуалізації таких векторних текстур. Особливістю рішення задачі є поєднання методу сферичної інтерполяції і синтезу зображення векторної текстури зворотним трасуванням за один прохід. Такий підхід дозволяє розширити області спільного застосування цих методів, в тому числі, на основі використання існуючих баз даних векторних текстур, заданих прямими лініями.

**Ключові слова:** моделювання кривих и поверхонь, квадрик, інтерполяція, сіткова модель.

### SYNTHESIS OF REVERSE TRACING VECTOR TEXTURE IMAGES FORMED SPHERICAL INTERPOLATION

*V. M. Gusiatin, M. V. Gusiatin*

In article questions of increase of realness of the scenes synthesized by method of the reverse tracing are considered. The task of visualization is solved by the reverse tracing of the vectorial textures set by straight lines and smoothed by method of spherical interpolation. Analytical ratios for implementation of each stage of visualization of such vectorial textures are given. Feature of the solution of the task is combination of a method of spherical interpolation and synthesis of the image of a vectorial texture the reverse tracing for one pass. This approach allows us to expand the joint application of these methods, including through the use of existing databases vector textures defined by straight lines

**Keywords:** method of the return tracing, modeling of curves and surfaces, quadrics, interpolation, texture.

**Гусятін Владимир Михайлович** – канд. техн. наук, доцент, професор каф. ЕВМ, Харківський національний університет радіоелектроніки, Харків, Україна, e-mail: [vmgus@rambler.ru](mailto:vmgus@rambler.ru).

**Гусятін Максим Владимирович** – інженер каф. ЕВМ, Харківський національний університет радіоелектроніки, Харків, Україна.

**Gusiatin Vladimir Mikhailovich** – Candidate of Technical Science, Professor, Department of Electronic Computers, Kharkov National University of Radio Electronics "NURE", Kharkov, Ukraine, e-mail: [vmgus@rambler.ru](mailto:vmgus@rambler.ru).

**Gusiatin Mixim Vladimirovich** - Engineer, Department of Electronic Computers, Kharkov National University of Radio Electronics "NURE", Kharkov, Ukraine.