

# Итеративный алгоритм восстановления трехмерных сцен, движения и фокусного расстояния камеры в перспективной проекции, основанный на факторизации матриц.

Н. В. Янова, Д. В. Юрин  
ЦОС и ВТ МФТИ  
Москва, Россия

## Аннотация

Разработан итерационный алгоритм точного решения нелинейной задачи восстановления трёхмерных сцен, движения, и внутренних параметров камеры, основанный на уточнении фокусного расстояния камер. Метод позволяет устранить неоднозначность восстановления сцены по третьей координате  $z$  в случае, когда перспективные искажения существенны. В предложенном алгоритме итерационно применяется приближение масштабируемой ортографической проекции [1]. Фокусные расстояния уточняются методом минимизации отношения сингулярных чисел масштабируемой матрицы измерений. Алгоритм обеспечивает лучшую точность по сравнению с линейными методами, и отличается от известных в настоящее время алгоритмов решения нелинейной задачи простотой, широким диапазоном применимости и высокой скоростью сходимости.

**Ключевые слова:** Восстановление трехмерных сцен, Метод факторизации.

## 1. ВВЕДЕНИЕ

В настоящее время задаче реконструкции трехмерных сцен посвящено большое количество исследований [1-5]. Разработано много различных подходов, основанных на различных физических принципах, таких, как восстановление формы из фокусировки и дефокусировки, стерео [4], угловой зависимости отражательных свойств материалов, рассеянию в атмосфере [5] и др. В настоящей работе за основу реконструкции трёхмерных сцен были взяты алгоритмы восстановления сцены по последовательности изображений, полученных с различных позиций одной или нескольких камер [3]. Среди них особый интерес представляют алгоритмы, основаны на факторизации матриц [1,2] в виду их вычислительной эффективности. Такие методы основаны на поиске характеристических точек на изображениях в виде уголков или линий [6,7]. Задача нахождения взаимнооднозначного соответствия между ними на различных изображениях обрабатываемой последовательности обычно решается методами траекторного анализа и калмановской фильтрации [3]. Нужно заметить, что методы [1-3] не работают непосредственно с изображениями, а требуют на вход набор координат характеристических точек изображений в пикселях, и наличия у каждой такой точки маркера (номера), причём, на всей последовательности изображений одной и той же точке реальной сцены должен соответствовать одинаковый маркер. В [1] наиболее компактно и подробно описаны различные приближения, а также рассматривается применение метода факторизации в тех случаях,

когда не все характеристические точки присутствуют на всех кадрах последовательности.

## 2. ПОСТАНОВКА ЗАДАЧИ

Рассмотрим систему координат  $(\mathbf{i}_f, \mathbf{j}_f, \mathbf{k}_f)$ , связанную с камерой, такую, что орт  $\mathbf{k}_f$  направлен вдоль оптической оси в направлении наблюдаемой сцены (см. рис. 1.).

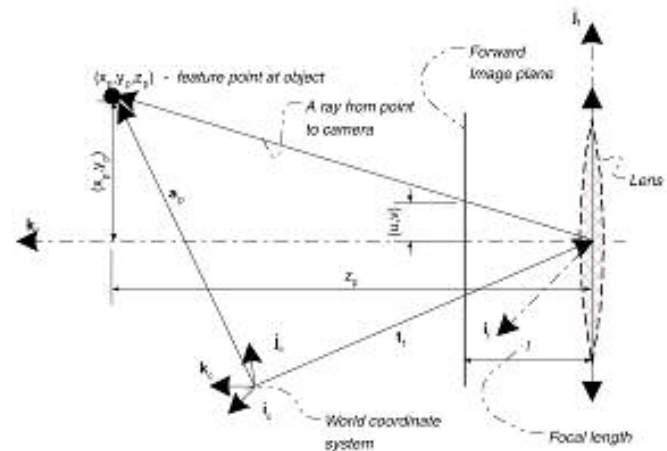


Рисунок 1: Постановка задачи для одной камеры

Векторы  $\mathbf{i}_f, \mathbf{j}_f, \mathbf{k}_f$  образуют правую ортонормированную тройку. Отвлекаясь от эффектов, связанных с ограниченной глубиной резкости, характерной для реальных оптических систем, будем считать, что изображения всех точек сцены в плоскости изображения находятся в фокусе, что, в частности, реализуется для камеры Обскура. Введем понятие передней плоскости изображения (ППИ), которая расположена в плоскости  $z=l$ , где  $l$  – фокусное расстояние объектива. В дальнейшем будем считать, что изображение формируется на ППИ. Это не влияет на описание задачи в рамках геометрической оптики, однако, позволяет избавиться от несущественных знаков и усложнения, связанного с лишним преобразованием системы координат. Орты  $\mathbf{i}_f, \mathbf{j}_f$  направлены соответственно вдоль строки и столбца пикселей изображения, формируемого в ППИ камеры. Пусть на объекте находится точка  $sp$  с координатами  $(x, y, z)$ . Координаты ее изображения на ППИ обозначим  $(\tilde{u}, \tilde{v})$ , тогда из подобия треугольников можно записать:

$$\frac{x}{z} = \frac{\tilde{u}}{l}, \quad \frac{y}{z} = \frac{\tilde{v}}{l} \quad (2.1)$$

Уравнения (2.1) устанавливают связь между измеряемыми на опыте значениями  $(\tilde{u}, \tilde{v})$ , координатами точки объекта в трехмерном пространстве и фокусным расстоянием  $l$  камеры.

Пусть имеется  $P$  точек на объекте и  $F$  камер или кадров, снятых при различных положениях камеры. Принадлежность величины к точке объекта  $p$  (point) будем обозначать индексом  $p=1, \dots, P$ . Индексом  $f$  (frame) будем обозначать величины, относящиеся к определенному кадру (камере и ее положению). Тогда уравнения (2.1) принимают вид:

$$\frac{x_{fp}}{z_{fp}} = \frac{\tilde{u}_{fp}}{l_f}, \quad \frac{y_{fp}}{z_{fp}} = \frac{\tilde{v}_{fp}}{l_f} \quad (2.2)$$

В практике работы с цифровыми камерами, размер изображения в ППИ (то есть, размер пикселя), как правило, в единицах длины неизвестен, и непосредственно не измеряется. Поэтому целесообразно перейти к измеримым величинам, таким, как координаты пикселя. Преобразуем уравнения (2.2) к виду:

$$\begin{cases} \tilde{u}_{fp} \frac{a}{\tilde{u}_f^e N} = \frac{x_{fp}}{z_{fp}} \frac{l_f a}{\tilde{u}_f^e N} \\ \tilde{v}_{fp} \frac{a}{\tilde{v}_f^e N} = \frac{y_{fp}}{z_{fp}} \frac{l_f a}{\tilde{v}_f^e N} \end{cases} \quad (2.3)$$

и перейдем к новым переменным:

$$\begin{cases} u_{fp} = \frac{a}{N} \frac{\tilde{u}_{fp}}{\tilde{u}_f^e} = a \frac{n_{fp}^{(x)}}{N} \\ v_{fp} = \frac{a}{N} \frac{\tilde{v}_{fp}}{\tilde{v}_f^e} = a \frac{n_{fp}^{(y)}}{N} \\ g_f = a \frac{l_f}{\tilde{u}_f^e N} = \frac{1}{\varphi_f} \\ \alpha_f = \frac{\tilde{u}_f^e}{\tilde{v}_f^e} \end{cases} \quad (2.4)$$

где  $\alpha_f$  - aspect ratio; величины  $n_{fp}^{(x)}, n_{fp}^{(y)}$  - это координаты пикселя на цифровом изображении;  $\varphi = 2tg(\beta_{\max} / 2)$ , где  $\beta_{\max}$  - максимальный угол зрения камеры.

Тогда, в новых обозначениях:

$$\begin{cases} u_{fp} = g_f \frac{x_{fp}}{z_{fp}} \\ v_{fp} = \alpha_f g_f \frac{y_{fp}}{z_{fp}} \end{cases} \quad (2.5)$$

В (2.3) были введены следующие обозначения:  $a$  - положительный масштабирующий коэффициент, отражающий диапазон изменения  $u_{fp}, v_{fp} : [-a/2, a/2]$ , его значение существенно только для устойчивости и точности численных методов;  $\tilde{u}_f^e, \tilde{v}_f^e$  - размеры одного пикселя на фоточувствительной матрице в метрических единицах;  $N = \max(N_x, N_y)$ , где  $N_x, N_y$  - ширина и высота изображения в пикселях, обычно  $N_x > N_y$ .

Переходя к произвольной системе координат, и записывая уравнения (2.2) в векторной форме, получим:

$$\begin{cases} u_{fp} = g_f \frac{\mathbf{i}_f(\mathbf{s}_p - \mathbf{t}_f)}{z_{fp}} \\ v_{fp} = \alpha_f g_f \frac{\mathbf{j}_f(\mathbf{s}_p - \mathbf{t}_f)}{z_{fp}} \\ z_{fp} = \mathbf{k}_f(\mathbf{s}_p - \mathbf{t}_f) \end{cases} \quad (2.6)$$

Уравнения (2.6) являются основой для решения задачи восстановления трехмерной формы объекта. Рассмотрим их подробнее. При съемке объекта измеряются величины  $u_{fp}, v_{fp}$  - всего 2FP величин. Обычно значения  $\alpha_f$  известны и равны единице, поэтому в дальнейшем будем этот параметр опускать. Неизвестными являются  $\mathbf{s}_p, \mathbf{t}_f, g_f$  - 3P+3F+F=3P+4F величин и F троек векторов  $(\mathbf{i}_f, \mathbf{j}_f, \mathbf{k}_f)$ , на которые наложены ограничения ортонормированности и правой тройки:

$$\begin{cases} \mathbf{i}_f \cdot \mathbf{i}_f = 1 \\ \mathbf{j}_f \cdot \mathbf{j}_f = 1 \\ \mathbf{i}_f \cdot \mathbf{j}_f = 0 \\ \mathbf{k}_f = \mathbf{i}_f \times \mathbf{j}_f \end{cases}, \quad f = 1, \dots, F \quad (2.7)$$

Итого получаем  $(6-3)F=3F$  неизвестных величин. Таким образом, 2FP уравнений определяют 3P+7F неизвестных. Система уравнений (2.6) может быть разрешена относительно этих неизвестных в смысле метода наименьших квадратов, если выполняется условие  $2FP > 3P+7F$ , которое при достаточном количестве точек объекта и снятых кадров может быть удовлетворено. Минимальными значениями  $F, P$  являются  $(F=5, P=5)$ ,  $(F=12, P=4)$ ,  $(F=3, P=7)$ . Заметим, что количество неизвестных может быть уменьшено, если заданы дополнительные условия, например, постоянство фокусного расстояния (отсутствие zoom-a), уменьшает число неизвестных на  $F-1$ . Важно отметить также, что размеры трехмерной сцены на базе уравнений (2.1) могут быть восстановлены только с точностью до масштабирующего множителя, поэтому для восстановления в абсолютных единицах требуется знание какого-либо размера сцены - расстояния между двумя точками, или расстояния между двумя положениями камеры.

### 3. МЕТОД ФАКТОРИЗАЦИИ

Выберем начало МСК в центре масс (ЦМ) точек объекта:

$$\frac{1}{P} \sum_{p=1}^P \mathbf{s}_p = 0 \quad (3.1)$$

Тогда

$$\begin{aligned} \frac{1}{P} \sum_{p=1}^P z_{fp} &= \mathbf{k}_f \frac{1}{P} \sum_{p=1}^P \mathbf{s}_p - \mathbf{k}_f \mathbf{t}_f = -\mathbf{k}_f \mathbf{t}_f = z_f, \\ z_{fp} &= z_f + \mathbf{k}_f \mathbf{s}_p = z_f \left(1 + \frac{\mathbf{k}_f \mathbf{s}_p}{z_f}\right). \end{aligned} \quad (3.2)$$

В приближении  $|\mathbf{s}_p| \ll z_f$ , уравнения (2.6) приводятся, как показано в [1], к масштабируемой ортографической проекции (МОП):

$$\begin{cases} u_{fp} = g \frac{\mathbf{i}_f(\mathbf{s}_p - \mathbf{t}_f)}{z_f} \\ v_{fp} = \alpha g \frac{\mathbf{j}_f(\mathbf{s}_p - \mathbf{t}_f)}{z_f} \\ z_f = -\mathbf{k}_f \mathbf{t}_f \end{cases} \quad (3.3)$$

Введя соответствующие обозначения, уравнения для МОП можно свести к виду:

$$\begin{cases} u_{fp} = \mathbf{m}_f \mathbf{s}_p + x_f \\ u_{fp} = \mathbf{n}_f \mathbf{s}_p + y_f \end{cases}, \text{ где } \begin{cases} \mathbf{m}_f = \mathbf{i}_f / z_f' \\ \mathbf{n}_f = \mathbf{j}_f / z_f' \end{cases} \quad (3.4)$$

$$\begin{cases} x_f = -\mathbf{i}_f \mathbf{t}_f / z_f' \\ y_f = -\mathbf{j}_f \mathbf{t}_f / z_f' \\ z_f' = -\mathbf{k}_f \mathbf{t}_f / g \end{cases}$$

Геометрический смысл МОП проиллюстрирован в [1].

Одним из наиболее эффективных методов восстановления трёхмерных форм и движения объекта, является метод, основанный на факторизации матриц. Поясним суть этого метода.

Сформируем из координат точек изображения  $\{(u_{fp}, v_{fp}) : f = \overline{1, F}, p = \overline{1, P}\}$  матрицу измерений  $\mathbf{W}$  размера  $2F \times P$ :

$$\mathbf{W} = \begin{pmatrix} u_{11} & u_{12} & \dots & u_{1P} \\ v_{11} & v_{12} & \dots & v_{1P} \\ \vdots & \vdots & \dots & \vdots \\ u_{F1} & u_{F2} & \dots & u_{FP} \\ v_{F1} & v_{F2} & \dots & v_{FP} \end{pmatrix} \quad (3.5)$$

Каждая строка матрицы  $\mathbf{W}$  содержит координаты точек  $(u_{fp}, v_{fp})$ , относящихся к определённому кадру последовательности, а каждый столбец, - те же величины для конкретной точки, присутствующей на всех изображениях. Тогда (3.4) можно переписать в матричной форме:

$$\mathbf{W} = \mathbf{M}\mathbf{S} + \mathbf{T}[1 \dots 1], \quad (3.6)$$

где  $\mathbf{M}$  – матрица движения размера  $2F \times 3$ ,  $\mathbf{S}$  – матрица формы размера  $3 \times P$ ,  $\mathbf{T}$  – вектор смещений камер, формирующиеся следующим образом:

$$\mathbf{T} = \begin{pmatrix} x_1 \\ y_1 \\ \vdots \\ x_F \\ y_F \end{pmatrix}, \quad \mathbf{S} = \|\mathbf{s}_1 \dots \mathbf{s}_P\|, \quad \mathbf{M} = \begin{pmatrix} \mathbf{m}_1^T \\ \mathbf{n}_1^T \\ \vdots \\ \mathbf{m}_F^T \\ \mathbf{n}_F^T \end{pmatrix}, \quad (3.7)$$

Учитывая условие (3.1), вектор  $\mathbf{T}$  получается построчным суммированием матрицы  $\mathbf{W}$ :

$$\mathbf{T}_f = \frac{1}{P} \sum_{p=1}^P w_{fp} \quad (3.8)$$

Поэтому далее удобно работать с матрицей  $\mathbf{W}^*$ :

$$\mathbf{W}^* = \mathbf{W} - \mathbf{T}[1 \dots 1] = \mathbf{M}\mathbf{S}. \quad (3.9)$$

Метод факторизации [1] основан на том, что, поскольку матрица  $\mathbf{W}^*$  представима в виде произведения двух матриц ранга 3, её ранг не может быть больше 3. Факторизация матрицы  $\mathbf{W}^*$  ранга 3 производится посредством сингулярного разложения матриц (SVD) с последующим занулением всех сингулярных чисел, кроме трех наибольших:

$$\mathbf{W}^* = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T = (\mathbf{U}\sqrt{\mathbf{\Sigma}})(\sqrt{\mathbf{\Sigma}}\mathbf{V}^T) = \tilde{\mathbf{M}}\tilde{\mathbf{S}}. \quad (3.10)$$

Здесь  $\mathbf{U}$ ,  $\mathbf{V}$  – ортогональные матрицы, а  $\mathbf{\Sigma} = \text{diag}(\sigma_1 \dots \sigma_n)$ ,  $n = \min(2F, P)$ . Понятно, что такое разложение не единственно, поскольку между матрицами  $\tilde{\mathbf{M}}$ ,  $\tilde{\mathbf{S}}$  можно поместить произведение прямой и обратной матриц  $\mathbf{Q}$  ранга 3, ничего при этом не изменив:

$$\mathbf{W}^* = \tilde{\mathbf{M}}\tilde{\mathbf{S}} = \tilde{\mathbf{M}}(\mathbf{Q}\mathbf{Q}^{-1})\tilde{\mathbf{S}} = (\tilde{\mathbf{M}}\mathbf{Q})(\mathbf{Q}^{-1}\tilde{\mathbf{S}}) = \mathbf{M}\mathbf{S}.$$

Так как  $\mathbf{M}^T\mathbf{M} = \tilde{\mathbf{M}}^T\mathbf{Q}^T\mathbf{Q}\tilde{\mathbf{M}}$ , то, в силу ортонормированности базисных векторов, задающих систему координат, связанную с камерой:

$$\begin{cases} |\mathbf{m}_f| = \frac{g}{z_f} \\ |\mathbf{n}_f| = \frac{g}{z_f} \\ \mathbf{m}_f \mathbf{n}_f = 0 \end{cases}, \quad (3.11)$$

получаем линейную систему уравнений [1] для нахождения шести неизвестных элементов матрицы  $\mathbf{Q}^T\mathbf{Q}$ :

$$\begin{cases} \tilde{\mathbf{m}}_f^T \mathbf{Q}^T \mathbf{Q} \tilde{\mathbf{m}}_f - \tilde{\mathbf{n}}_f^T \mathbf{Q}^T \mathbf{Q} \tilde{\mathbf{n}}_f = 0 \\ \tilde{\mathbf{m}}_f^T \mathbf{Q}^T \mathbf{Q} \tilde{\mathbf{n}}_f = 0 \\ \tilde{\mathbf{m}}_1^T \mathbf{Q}^T \mathbf{Q} \tilde{\mathbf{m}}_1 = 1 \end{cases} \quad (3.12)$$

В предлагаемой работе система уравнений (3.12) решалась методом наименьших квадратов с применением SVD. Элементы матрицы  $\mathbf{Q}$  находятся путем факторизации матрицы  $\mathbf{Q}^T \mathbf{Q}$ . Заметим, что уравнения этой системы остаются справедливыми при умножении в (3.12) матрицы  $\mathbf{Q}$  на матрицу вида:

$$\begin{pmatrix} \pm 1 & 0 & 0 \\ 0 & \pm 1 & 0 \\ 0 & 0 & \pm 1 \end{pmatrix} \quad (3.13)$$

Это обусловлено тем, что до сих пор было зафиксировано только начало координат МСК, а ориентация осей в процессе решения уравнений (3.10)-(3.12) оставалась произвольна. Зафиксировав ориентацию осей системы координат, например, выбрав направление осей таким же, как у системы, связанной с первой камерой, неоднозначность в первых двух знаках (3.13) можно устранить. Неоднозначность в третьем знаке (3.13) связана с тем, что в МОП пренебрегают глубиной объекта по сравнению с расстоянием до него [1]. Поэтому алгоритмы, основанные на МОП, работают, по существу, с плоскими объектами. Это приводит к неоднозначности восстановления формы сцены  $\mathbf{S}$  и движения камер  $\mathbf{M}$ .

Учитывая выравнивание по первой камере, получим искомые матрицы движения и формы:

$$\begin{cases} \mathbf{M}' = \mathbf{M}\mathbf{R}_0 \\ \mathbf{S}' = \mathbf{R}_0^T \mathbf{S} \\ \mathbf{R}_0 = [\mathbf{i}_1, \mathbf{j}_1, \mathbf{k}_1] \end{cases} \quad (3.14)$$

#### 4. ИТЕРАЦИОННЫЙ АЛГОРИТМ ТОЧНОГО РЕШЕНИЯ ЗАДАЧИ ВОССТАНОВЛЕНИЯ

Основными недостатками рассмотренного выше приближения МОП являются неоднозначность в определении знака глубины сцены и невозможность вычисления фокусных расстояний камер. Наличие на изображениях перспективных искажений является дополнительным источником информации, позволяющим устранить перечисленные недостатки.

Уравнения (2.7) можно переписать в следующем виде:

$$\begin{cases} (1 + \frac{\mathbf{k}_f \mathbf{s}_p}{z_f}) u_{fp} = \frac{g}{z_f} (\mathbf{i}_f \mathbf{s}_p - \mathbf{i}_f \mathbf{t}_f) \\ (1 + \frac{\mathbf{k}_f \mathbf{s}_p}{z_f}) v_{fp} = \frac{g}{z_f} (\mathbf{j}_f \mathbf{s}_p + \mathbf{j}_f \mathbf{t}_f) \end{cases} \quad (4.1)$$

Введем новые обозначения:

$$\begin{cases} z'_f = z_f/g, & \begin{cases} x_f = -\mathbf{i}_f \mathbf{t}_f / z'_f \\ y_f = -\mathbf{j}_f \mathbf{t}_f / z'_f \end{cases} \\ \alpha = 1/g, & \\ \begin{cases} \mathbf{m}_f = \mathbf{i}_f / z'_f \\ \mathbf{n}_f = \mathbf{j}_f / z'_f \end{cases} & \begin{cases} u'_{fp} = (1 + \alpha \mathbf{k}_f \mathbf{s} / z'_f) u_{fp} \\ v'_{fp} = (1 + \alpha \mathbf{k}_f \mathbf{s} / z'_f) v_{fp} \end{cases} \end{cases} \quad (4.2)$$

Тогда формулы для перспективной проекции (4.1) принимают вид, соответствующий МОП - приближению:

$$\begin{cases} u'_{fp} = \mathbf{m}_f \mathbf{s}_p + x_f \\ v'_{fp} = \mathbf{n}_f \mathbf{s}_p + y_f \end{cases} \quad (4.3)$$

Уравнения (4.3) совпадают с уравнениями (3.4), за исключением того, что их левая часть, наряду с данными измерений, содержит теперь и неизвестные величины  $s_p, k_f$ . В матричной записи они принимают вид, аналогичный (3.6), где матрица  $\mathbf{W}$  может быть представлена в виде двух слагаемых, первое из которых,  $\mathbf{W}_1$ , соответствует (3.5), а второе,  $\mathbf{W}_2$ , зависит от  $\mathbf{W}_1, s_p, k_f$  и имеет смысл поправки на перспективные искажения:

$$\mathbf{W} = \mathbf{M}\mathbf{S} + \mathbf{T}, \quad \text{где} \quad \mathbf{W} = \mathbf{W}_1 + \alpha \mathbf{W}_2 \quad (4.4)$$

Система уравнений (4.4) может быть решена итерациями, путём уточнения этой поправки. Заметим здесь, что, в отличие от алгоритма, предложенного в [2], количество параметров, подбираемых по итерациям (фокусные расстояния камер) существенно меньше, -  $F$  величин в предлагаемом алгоритме, и  $FP$  величин в [4], а структура каждой строки матрицы  $\mathbf{W}_2$  предварительно определяется из приближенного решения. Сформулируем предлагаемый алгоритм решения нелинейной задачи восстановления:

1. Полагаем в (4.4):

$$q := 0, \quad \alpha^{(0)} := 0, \quad \mathbf{W}^{(0)} := \mathbf{W}_1, \quad \mathbf{W}_2^{(0)} := 0$$

2. Решаем систему уравнений, описывающую МОП - приближение:

$$\mathbf{W}^{(q)} = \mathbf{M}^{(q)} \mathbf{S}^{(q)} + \mathbf{T}^{(q)}.$$

3. Полагаем теперь:

$$q := q + 1$$

4. Вычисляем матрицу

$$\mathbf{W}_2^{(q)} := \mathbf{W}_2^{(q)} (\mathbf{M}^{(q-1)}, \mathbf{S}^{(q-1)}, \mathbf{T}^{(q-1)}).$$

5. Находим параметр  $\alpha^{(q)}$ , подбирая его таким, чтобы ранг матрицы  $\mathbf{W}$  оставался равным 3, то есть надо искать:

$$\min_{\alpha^{(q)}} \frac{\sigma_4}{\sigma_1} (\mathbf{W}_1 + \alpha^{(q)} \mathbf{W}_2^{(q)}),$$

где  $\sigma_1, \sigma_4$  - сингулярные числа матрицы

$\mathbf{W}^{(q)} : \mathbf{W}^{(q)} \stackrel{SVD}{=} \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T, \mathbf{\Sigma} = \text{diag}(\sigma_1 \dots \sigma_4)$ . Диапазон изменения  $\alpha^{(q)}$  определяется из условия:

$$1 + \alpha^{(q)} \frac{\mathbf{k}_f \mathbf{s}_p}{z'_f} > 0.$$

6. Вычислить матрицу  $\mathbf{W}^{(q)} := \mathbf{W}_1 + \alpha^{(q)} \mathbf{W}_2^{(q)}$ .

7. Перейти к п. 2, если не выполняется условие:

$$|\alpha^{(q-1)} - \alpha^{(q)}| < \varepsilon.$$

8. Устранить неоднозначность знака глубины объекта: Если  $\alpha^{(q)} < 0$ ,

$$\alpha^{(q)} := -\alpha^{(q)} \quad \text{то}$$

$$\mathbf{S}^{(q)} := Q_1 \mathbf{S}^{(q)} \quad , \quad \text{где} \quad Q_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

$$\mathbf{M}^{(q)} := Q_1 \mathbf{M}^{(q)}$$

Сделать выравнивание системы координат по первой камере (3.14).

Вычислить координаты позиций камер.

## 5. ПОСТРОЕНИЕ МОДЕЛИ ДЛЯ ГЕНЕРАЦИИ СИНТЕТИЧЕСКИХ ДАННЫХ

В среде Matlab была разработана модель, представляющая собой три грани прямоугольного параллелепипеда, на каждой из которых характеристические точки образуют равномерную прямоугольную сетку. С помощью построенной модели был организован автоматический генератор характеристических точек, который необходим в процессе реконструкции трёхмерных сцен, поскольку алгоритмы, как правило, зависят от числа характеристических точек. Помимо этого, появилась возможность сравнения полученных результатов восстановления с эталоном. В модели в качестве характеристических точек берутся координаты вершин прямоугольников, принадлежащих поверхности трёхгранного угла. Характеристические точки хранятся в матрице  $\mathbf{X}$  размера  $N_{\text{points}} \times 3$ , в процессе формирования которой было учтено, что на пересечении рёбер трёхгранного угла лежат одни и те же точки.

В качестве одного из методов оценки результатов восстановления использовался язык Virtual Reality Modeling Language (VRML). Для представления результатов в среде VRML требуется задание не только трёхмерных координат точек, но и рёбер куба. Поэтому, параллельно с построением матрицы  $\mathbf{X}$ , запоминается способ обхода последовательности точек на грани, посредством их нумерации. Рёбра куба представляются матрицей **FaceSet**, в которую записывается последовательность маркеров точек в вершинах обходённых прямоугольников. В модели введены 3 системы координат: объектная система координат (ОСК), с центром в вершине трёхгранного угла, МСК с началом в ЦМ точек объекта, и система координат, связанная с камерой (КСК). Все камеры позиционированы таким образом, чтобы объект всегда находился в поле зрения каждой камеры, и сориентированы на объект следующим способом:

$$\begin{cases} \mathbf{i}_{cam} = \frac{\mathbf{j}_{obj} \times \mathbf{k}_{cam}}{|\mathbf{j}_{obj} \times \mathbf{k}_{cam}|} \\ \mathbf{j}_{cam} = -\mathbf{i}_{cam} \times \mathbf{k}_{cam} \\ \mathbf{k}_{cam} = \frac{\mathbf{x}_C - \mathbf{x}_{Cam}}{|\mathbf{x}_C - \mathbf{x}_{Cam}|} \end{cases} \quad (5.1)$$

Здесь векторы  $\mathbf{x}_C$ ,  $\mathbf{x}_{Cam}$  задают положение ЦМ объекта и позиции камер, ОСК определяется следующим образом:

$$\begin{cases} \mathbf{i}_{obj} = [100]^T \\ \mathbf{j}_{obj} = [010]^T \\ \mathbf{k}_{obj} = [001]^T \end{cases} \quad (5.2)$$

В модели также предусмотрено, что камеры могут наводиться не только на ЦМ точек объекта, но и на произвольную точку. В модели камеры представляются матрицей **CamPos** размера  $3 \times N_{cam}$ , формируемой из векторов смещений всех камер. Предусмотрена также возможность совмещения исходной модели с данными, полученными в результате восстановления.

## 6. РЕЗУЛЬТАТЫ

Предложенный алгоритм тестировался на реальных и синтетических данных. В настоящей работе приводится сопоставление результатов восстановления в приближении масштабируемой ортографической проекции [1] и предлагаемым методом. Результаты восстановления сопоставляются также с исходной моделью. Так как МОП не восстанавливает фокусные расстояния камер, то, при сравнении этого приближения с моделью, использовались истинные значения фокусных расстояний, полученные из модели.

Для всех представленных в настоящей работе результатов, расстояние от ближайшей к объекту камеры до его центра масс  $\mathbf{r}_{\min}$  варьировалось в диапазоне от 2 до 30 размеров объекта. В каждой последовательности расстояние от центра масс объекта до самой удаленной камеры  $\mathbf{r}_{\max}$  в 1.5 раза превышало расстояние до ближайшей камеры.

На рис. 2-3 представлены результаты восстановления формы объекта и положений камер, в сравнении с моделью для МОП - приближения, и предлагаемого метода. Истинные положения камер изображены кружочком, восстановленные – крестиком. Видно, что в присутствии сильных перспективных искажений предлагаемый метод обеспечивает существенно лучшую точность. На рис. 4-5 приведена аналогичная пара результатов для случая больших расстояний от камер до объекта, камеры не показаны, поскольку в такой ситуации объект на рисунке будет очень мал. Видно, что, даже в этих условиях, предлагаемый метод обеспечивает лучшую точность восстановления формы, однако точность восстановления положений камер несколько снижается из-за погрешностей восстановления фокусного расстояния. Заметим здесь, что при очень больших расстояниях, перспективные искажения отсутствуют, это и делает принципиально невозможной оценку фокусного расстояния предлагаемым методом.

В модели, описанной в предыдущем разделе, фокусные расстояния вычислялись таким образом, чтобы изображение объекта, полученное с ближайшей к объекту камеры, занимало большую часть площади кадра. На рис. 6 пунктиром обозначены значения фокусных расстояний, полученные из модели, сплошной линией - результаты восстановления предлагаемым алгоритмом, и точечным пунктиром – относительная ошибка в процентах. На рис. 7-9 результаты для МОП - приближения показаны штрих - пунктирной линией, а для предлагаемого метода – сплошной линией с круглыми маркерами.

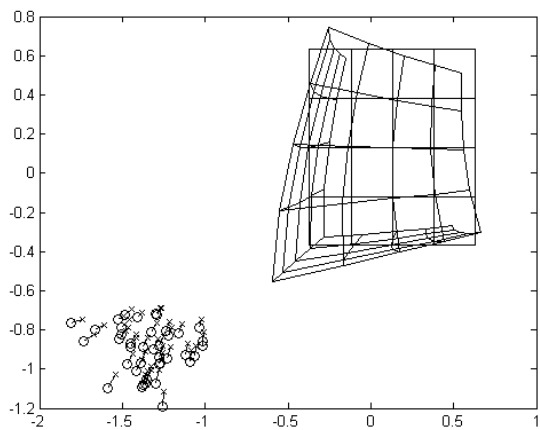


Рисунок 2: Результаты восстановления в МОП,  
 $r_{\min} = 2a$ .

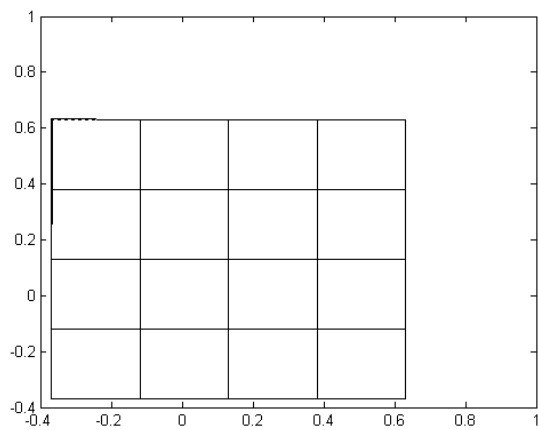


Рисунок 5: Результаты восстановления предлагаемым методом при слабых искажениях,  $r_{\min} = 30a$ .

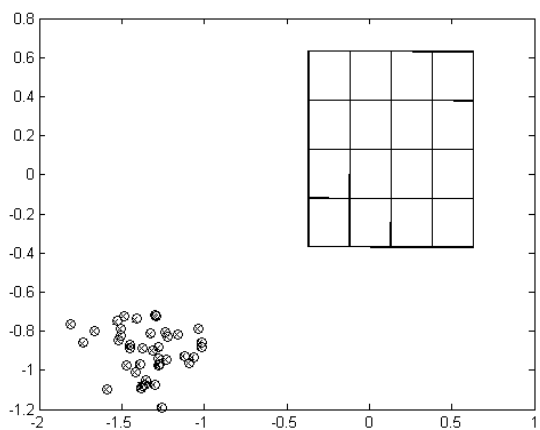


Рисунок 3: Результаты восстановления предлагаемым методом,  $r_{\min} = 2a$ .

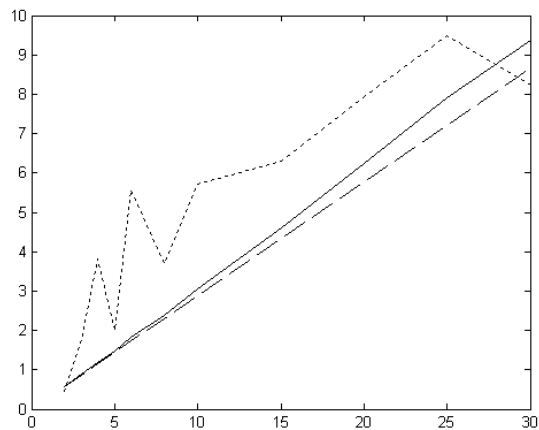


Рисунок 6: Погрешность восстановления фокусного расстояния камер в зависимости от  $r$  для данных, приведённых на рис.2-5.

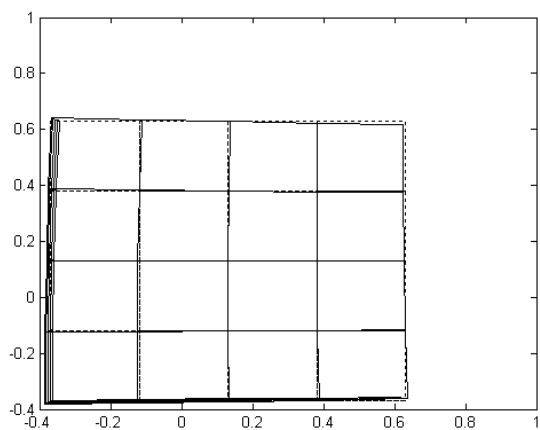


Рисунок 4: Результаты восстановления в МОП,  
 $r_{\min} = 30a$

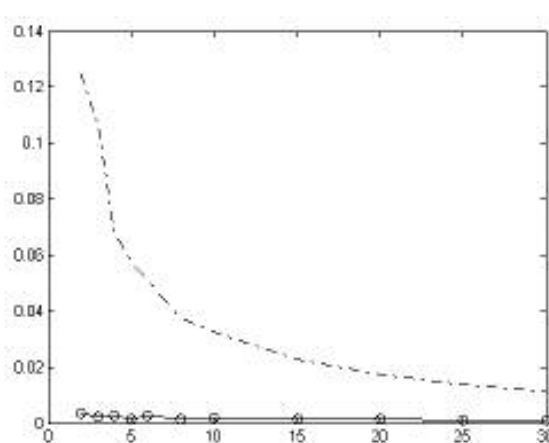


Рисунок 7: Погрешность восстановления формы сцены в зависимости от  $r$  для данных, приведённых на рис.2-5.

На рис. 7 приведена погрешность восстановления формы объекта, которая рассчитывалась как среднеквадратичное отклонение координат восстановленных точек от модели. На рис. 8 показана погрешность восстановления ориентации камер. Видно, что во всех случаях предлагаемый метод даёт лучшую точность. На рис. 9 проиллюстрирована погрешность восстановления позиций камер. Её рост с расстоянием для предлагаемого метода, обусловлен тем, что в нём использовались восстановленные значения фокусных расстояний (см. рис. 6), для МОП - приближения фокусные расстояния брались непосредственно из модели.

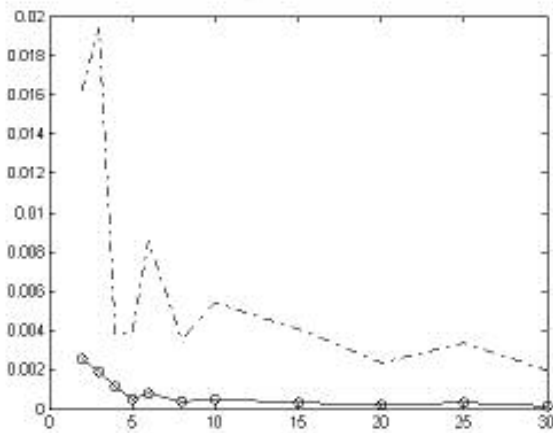


Рисунок 8: Погрешность восстановления ориентации камер в зависимости от  $r$  для данных, приведённых на рис.2-5.

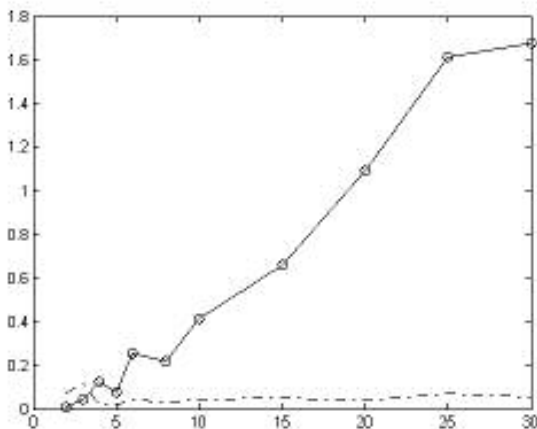


Рисунок 9: Погрешность восстановления позиций камер в зависимости от  $r$  для данных, приведённых на рис.2-5.

## 7. ЗАКЛЮЧЕНИЕ

Разработан итерационный алгоритм решения задачи восстановления трехмерных сцен, движения и внутренних параметров камер по последовательности изображений в перспективной проекции. Предлагаемый подход обеспечивает заметно более высокую точность, чем приближение масштабированной ортографической проекции [1] и позволяет восстанавли-

вать фокусные расстояния камер при наличии перспективных искажений, то есть при небольшом расстоянии от камер до объекта. С ростом этого расстояния погрешность восстановления фокусного расстояния растет, однако структура алгоритма такова, что позволяет легко модернизировать его для динамического переключения на линейные методы [1], на основании оценки фокусного расстояния в случае больших расстояний до объекта, где линейные методы обеспечивают хорошую точность. Систематический характер зависимости погрешности восстановления фокусного расстояния с удалением камер от объекта, позволяет предположить, что этот эффект может быть объяснен и частично скомпенсирован. Отметим здесь, что для сильно удаленных объектов перспективные искажения отсутствуют, и восстановление фокусных расстояний камер становится принципиально невозможным. Предлагаемый алгоритм, в отличие от линейных методов, не приводит к неоднозначностям восстановления трехмерной сцены [1]. В проведенных экспериментах алгоритм сходился за несколько итераций и показал высокую устойчивость по отношению к внесению шума в исходные данные.

## 8. ЛИТЕРАТУРА:

- [1] *Conrad I. Poelman, Takeo Kanade*. A Paraperspective Factorization Method for Shape and Motion Recovery: //Technical Report CMU-CS-93-219 / School of Computer Science, Carnegie Mellon University. — 11 December 1993.
- [2] *Mei Han, Takeo Kanade*. Scene Reconstruction from Multiple Uncalibrated Views //Technical Report CMU-RI-TR-00-09 / CMU Robotics Institute. — January 2000.
- [3] *Tony Jebara, Ali Azarbayejani and Alex Pentland*. [3D Structure from 2D Motion](#). //IEEE Signal Processing Magazine. May 1999, V.16, No.3. (MIT Media Laboratory. Perceptual Computing Technical Report #523).
- [4] *Sebastien Roy, Ingemar J. Cox*. A Maximum – Flow Formulation of the N-camera Stereo Correspondence Problem. // IEEE Proc. Of Int. Conference on Computer Vision, Bombai, January 1998.
- [5] *Fabio Cozman, Eric Krotkov*. Depth from Scattering. //Robotics Institute, Carnegie Mellon University, Pittsburgh. [http://www.ri.cmu.edu/pub\\_files/pub2/cozman\\_fabio\\_1997\\_1/cozman\\_fabio\\_1997\\_1.pdf](http://www.ri.cmu.edu/pub_files/pub2/cozman_fabio_1997_1/cozman_fabio_1997_1.pdf).
- [6] *S.M. Smith, J.M.Brady*. SUSAN - New approach to low-level image processing. //Int. journal of computer vision. Volume 23 No.1 P. 45-78, May 1997.
- [7] *Непомнящий П.В., Хельвас А.В., Юрин Д.В.* Обнаружение угловых структур на контурных изображениях полученных сегментацией раstra. //Обработка информации и моделирование. М., 2002 (в печати).

## Об авторах

Наталья Владимировна. Янова – студентка 5-го курса ФФКЭ Московского-Физико-Технического Института.

E-mail: [Natalie@cos.ru](mailto:Natalie@cos.ru)

Юрин Дмитрий Владимирович, к.ф.-м.н.

E-mail: [yurin@cos.ru](mailto:yurin@cos.ru)