ЕВРАЗИЙСКОЕ НАУЧНОЕ ОБЪЕДИНЕНИЕ
EURASIAN SCIENTIFIC ASSOCIATION

Теоретические основы построения трехмерной модели лица человека по фотографиям

Власов Константин Евгеньевич, студент
Новосибирский государственный технический университет

Определение координат вершин будущей трехмерной модели.

Задача построения трехмерной модели лица человека по фотографии сводится к тому, чтобы определить набор из трех пространственных координат (*x, y, z*) для каждой точки будущей модели. Однако все дело в том, что получить пару координат (*x, y*) для любой точки плоской фотографии не составляет труда, но фотография двухмерна, и мы не знаем «глубину» каждой точки. Если мы будем иметь для каждой точки третью координату, то уже сможем построить трехмерную модель, так как будем иметь массив вершин нашего трехмерного объекта.

Идея вычисления координаты *z* видимого плоского изображения исходит из идеи о том, как именно наш с вами мозг воспринимает трехмерные объекты. Технология проста: два абсолютно плоских изображения – проекций видимого пространства на сетчатку глаза – воспринимаются мозгом и обрабатываются одновременно, в результате чего формируется объемная модель наблюдаемой сцены и ее объектов. Наше с вами зрение называют бинокулярным, или иначе – *стереоскопическим*.

Данный метод основан на получении и обработке *стереопары фотографий*. После изучения реальных существующих методов построения трехмерной модели объекта по фотографиям было принято решения именно в пользу этого метода, поскольку в условиях ограниченного количества базовых фотографий он дает достаточно высокую скорость вычисления и хорошую точность.

На рисунке 1 представлен пример работы стереосистемы для получения стереопары изображений. Для точки S реальной модели существуют проекции на плоскость снимков – точки S_1 и S_2.

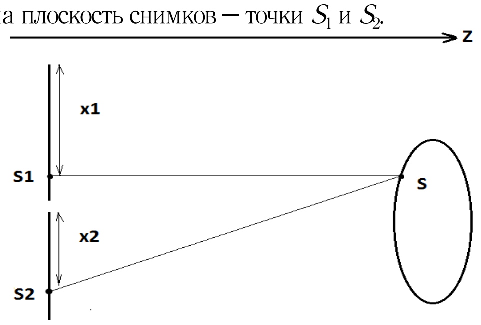

Рис. 1. Схематическое изображение стереосистемы

Итак, нам необходимо получить набор из координат *x, y* и *z* для точки *S*. В качестве пространственных координат (*x, y*) точек будущей трехмерной модели можно взять соответствующие координаты точек одной из фотографий. Для удобства одна из фотографий является фронтальной, то есть снимок делается под прямым углом к лицу.

Координата *z* любой точки при нормальной стереосъемке может быть вычислена с помощью координат x_1 и x_2 для этой точки на разных проекциях. Допустим, мы имеем некую точку S в пространстве, а ее проекции на плоскости фотографий имеют координаты $S_1=(x_1, y_1)$ и $S_2=(x_2, y_2)$. Тогда:

$$z = \frac{k}{x_1 - x_2} \quad (1.1)$$

где k – коэффициент, зависящий от системы.

Величина, вычисляемая в знаменателе данной дроби называется *диспаратностью*, от латинского слова *disparatus* – разделение. Диспаратность характеризует рассогласование двух точек на плоскости.

Другой вариант вычисления координаты *z* основан на вычислении сторон прямоугольных треугольников при известных углах.

Допустим, у нас есть некоторая точка *A* и две ее проекции на оси x_1 и x_2 (рис. 2).

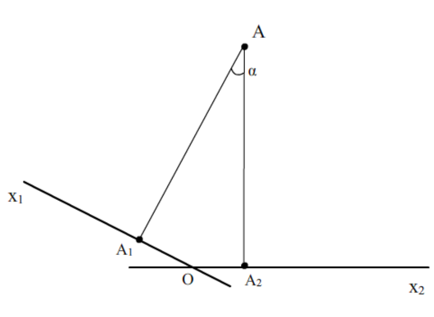

Рис. 2. Точка *A* и ее проекции на две оси

Здесь α – это угол между нормалями к осям, точки A_1 и A_2 – проекции точки *A* на соответствующие оси. Так же с точки зрения системы «фотоаппарат-объект» α – угол поворота (смещения) камеры относительно объекта фотосъемки для получения второго снимка.

Итак, построим на нашем рисунке ось y, параллельную отрезку AA_2, проходящую через точку пересечения осей x_1 и x_2 (рис. 3).

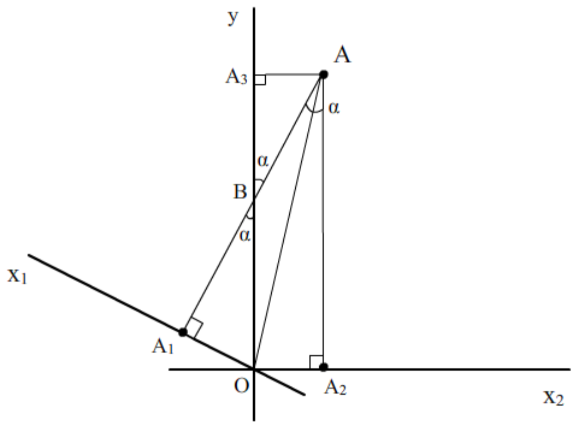

Рис. 3. Дополнительные построения

На рисунке выше точка A_3 – это проекция точи A на ось y, а B – это точка пересечения оси y и отрезка AA_1.

Теперь, чтобы определить расстояние до точки A от оси x_2 нам необходимо вычислить длину отрезка OA_3. Для этого необходимо вычислить длины отрезков OB и BA_3.

$$OA_3 = OB + BA_3 \ (1.2)$$

Начнем с того, что углы A_3BA и A_1AA_2 равны как накрест лежащие, а углы A_1BO и A_3BA равны как вертикальные.

$$\angle A_1BO = \angle A_3BA = \angle A_1AA_2 = \alpha \ (1.3)$$

Рассмотрим треугольники A_1OB и A_3AB. По теореме синусов для A_3AB справедливо:

$$\frac{A_3A}{\sin(\alpha)} = \frac{BA_3}{\sin(90^0 - \alpha)} \ (1.4)$$

Откуда получаем выражение для отрезка A_3B:

$$BA_3 = A_3A \cdot ctg(\alpha) \ (1.5)$$

Для треугольника A_1OB справедливо следующее равенство:

$$\frac{OB}{\sin(90^0)} = \frac{A_1O}{\sin(\alpha)} \ (1.6)$$

Откуда получаем выражение для отрезка OB:

$$OB = \frac{A_1O}{\sin(\alpha)} \ (1.7)$$

В итоге формула для расчета расстояния до точки A до оси x_2 принимает вид:

$$OA_3 = \frac{A_1O}{\sin(\alpha)} + \frac{A_3A}{\mathrm{tg}(\alpha)} \ (1.8)$$

С помощью полученной формулы можно вычислить координату z нашей будущей модели при условии, что мы знаем координаты x_1 и x_2 проекции точки, а так же угол α. Причем угол, конечно, можно задавать примерно, поскольку влияет он только лишь на величину z. Однако для достижения большей точности желательно замерять угол α. Для этого при съемке стереофотографий достаточно использовать горизонтальную поверхность и транспортир.

Определение контрольных точек.

Как было упомянуто в предыдущем параграфе, для построения точки в пространстве мы вычисляем координату z, имея две проекции этой точки. Таким образом, теперь задача сводится к определению такой пары точек $A_1(x_1, y_1)$ и $A_2(x_2, y_2)$ на стереофотографиях, чтобы они являлись проекциями одной точки A реального объекта.

Даная задача ложится на плечи пользователя приложения. С помощью простого инструмента пользователь должен отметить данные точки на фотографиях. В качестве проецируемых точек объекта выбираются так называемые *контрольные точки*. Контрольными точками называются те точки, которые определяют геометрию лица. Их может быть сколь угодно много, поскольку любая точка объекта является частью геометрии, однако есть несколько основных с помощью которых можно определить примитивную геометрию лица.

Ниже определены основные контрольные точки и фрагменты лица:

1. Нос (Границы крыльев носа, кончик носа, переносица)
2. Рот (уголки губ, середина верхней губы)
3. Глаза (уголки глаз, зрачок)
4. Брови (крайние точки бровей)

Контрольные фрагменты – те фрагменты лица, которые являются не точками, а описываются некоторыми кривыми. В данном случае число контрольных точек, соответствующих каждому фрагменту, может быть любым, все определяется пользователем. Однако каждый из этих фрагментов может быть представлен только одной контрольной точкой. Например, верхнее веко можно обозначить точкой в его середине.

Важно так же осознавать, что проекции выбранных контрольных точек на стереофотографиях должны быть хорошо заметны. То есть если мы хотим выделить дополнительные точки, например – несколько точек на лбу или щеках, нужно выбирать такие, которым можно легко найти соответственные на другом изображении. Это могут быль хорошо видимые морщины, дефекты, точки на татуировках, родимые пятна и родинки. Однако в случае ошибки или если соответственные точки определяются «примерно», можно сильно исказить результирующую модель.

Создание поверхности. Триангуляция.

Задача получения массива трехмерных координат модели является основной, но ее решение не дает в результате полноценную модель. Для построения поверхности модели и последующего ее текстурирования необходимо объединить вершины в полигоны. Вот в этом и заключается основная проблема, поскольку массив вершин строится, что называется, «с нуля». Это значит, что для него пока не существует массива индексов, который бы указывал, как и какие вершины следует связывать в полигоны.

Для решения этой задачи существует такое понятие как *триангуляция*. Для данного понятия есть несколько определений, но нас интересует только конкретное.

Триангуляцией называется планарный граф, все внутренние области которого являются треугольниками. Задачей построения триангуляции по заданному набору двумерных точек называется задача соединения заданных точек непересекающимися отрезками так, чтобы образовалась триангуляция. То же самое справедливо и для множества трехмерных точек.

В триангуляции можно выделить три основных объекта: узел (точка, вершина), ребро (отрезок) и треугольник.

В число операций над объектами триангуляции входят операции с ребрами, треугольниками и вершинами. При этом помимо определения входящих в состав объекта подобъектов (как, например, получение группы вершин треугольника) существуют операции определения смежных треугольников для текущего треугольника, ребра или вершины.

Существует несколько структур данных триангуляции определяющихся способом задания и определения треугольников.

1. Двойные ребра. Основа триангуляции – список ориентированных ребер. При этом каждое ребро входит в структуру дважды. Представление треугольников в неявном виде через подобъекты.
2. Узлы и треугольники. Каждый треугольник «знает» о соседних треугольниках и хранит образующие его вершины. Ребра задаются в неявном виде.
3. Узлы, ребра и треугольники. Все объекты задаются в явном виде.
4. Узлы, простые ребра и треугольники. Все объекты задаются в явном виде, однако ребра «знают» лишь о своих вершинах, не храня информации, например, о смежных треугольниках или ребрах.

Важнейшей операцией при выполнении триангуляции является проверка *условия Делоне*. Условие Делоне гласит, что сфера, описанная вокруг симплекса не содержит вершин других симплексов из заданного набора триангуляции.

Существует несколько способов проверки условия. Кроме метода, основанного на прямой проверке через уравнение окружности, проходящей через вершины треугольника, существуют методы с лучшей производительностью. К таковым относится метод *проверки суммы противолежащих углов*.

Идея метода состоит в том, что для заданного треугольника с вершинами в точках (x_1, y_1), (x_2, y_2) и (x_3, y_3) условие Делоне будет выполняться только тогда, когда для точки (x_0, y_0), принадлежащей триангуляции, будет выполняться условие:

$$\alpha + \beta \leq \pi \ (2.1)$$

$$\sin(\alpha + \beta) \geq 0 \ (2.2)$$

Если мы развернем функцию синуса из (10.2) в соответствии с формулой синуса суммы, то получим следующее неравенство:

$$\sin(\alpha)\cos(\beta) + \cos(\alpha)\sin(\beta) \geq 0 \ (2.3)$$

Значения синусов вычислим через произведения векторов, векторные и скалярные:

$$\cos(\alpha) = \frac{(x_0 - x_1)(y_0 - y_3) + (x_0 - x_3)(y_0 - y_1)}{\sqrt{(x_0 - x_1)^2 + (y_0 - y_1)^2} \cdot \sqrt{(x_0 - x_3)^2 + (y_0 - y_3)^2}} \ (2.4)$$

$$\sin(\alpha) = \frac{(x_0 - x_1)(x_0 - x_3) - (y_0 - y_1)(y_0 - y_3)}{\sqrt{(x_0 - x_1)^2 + (y_0 - y_1)^2} \cdot \sqrt{(x_0 - x_3)^2 + (y_0 - y_3)^2}} \ (2.5)$$

По аналогии вычисляется для угла β. Далее, подставив полученные формулы в формулу (3.3) и проведя преобразования, получим:

$$\big((x_0 - x_1)(y_0 - y_3) - (x_0 - x_3)(y_0 - y_1)\big)\big((x_2 - x_1)(x_2 - x_3) + (y_2 - y_1)(y_2 - y_3)\big) + \big((x_0 - x_1)(x_0 - x_3) - (y_0 - y_1)(y_0 - y_3)\big)\big((x_2 - x_1)(y_2 - y_3) - (x_2 - x_3)(y_2 - y_1)\big) \geq 0 \ (3.6)$$

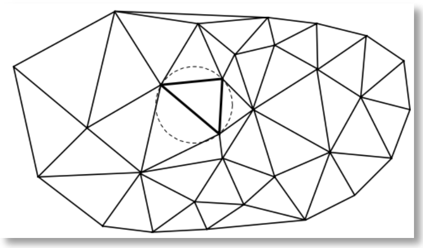

Рис. 4. Триангуляция Делоне

Проверка условия Делоне по заданной формуле в несколько раз сокращает количество простых операций (умножение и сложение) по сравнению с прямой проверкой на основании уравнения окружности.

В рамках поставленной задачи нас интересуют существующие алгоритмы построения триангуляции Делоне. Их можно разделить на четыре типа:

1. Итеративные алгоритмы. Представляют самую обширную группу алгоритмов. Итеративные алгоритмы несут под собой идею добавления точек в уже частично построенный граф;
2. Алгоритмы слияния. Для данных алгоритмов предполагается разбиение множества точек на подмножества, построение триангуляций на этих подмножествах, а затем объединение (слияние) триангуляций;
3. Алгоритмы прямого построения;
4. Двухпроходные алгоритмы.

Остановимся на итеративных алгоритмах. Итеративные алгоритмы имеют под собой идею добавления точек в уже частично построенный граф. Данные алгоритмы достаточно просты и вполне подходят для решения нашей задачи.

Существует жадный алгоритм построения триангуляции. Он был предложен одним из первых и выполняется всего в два этапа:

1. Генерируется список всех возможных отрезков и сортируется по возрастанию.
2. Начиная с самого короткого, происходит последовательное добавление ребер в триангуляцию. Так же происходит проверка, пересекается ли каждый новый отрезок с уже добавленными отрезками. В случае пересечения данное ребро выбрасывается из списка.

Данный алгоритм, не смотря на всю простоту, не подходит для решения текущей задачи, поскольку не позволяет хранить информацию о построенных треугольниках.

Итеративные алгоритмы включают четыре основные подгруппы:

1. Простой итеративный алгоритм. Алгоритм «удаляй и строй»;
2. Алгоритмы с индексированием поиска треугольников;
3. Алгоритмы с кэшированием поиска треугольников;
4. Алгоритмы с измененным порядком добавления точек.

Самый простой алгоритм построения триангуляции так и называется – простой итеративный алгоритм, или «удаляй и строй». Сама идея построения триангуляции учитывает перестраивание треугольников по необходимости. Однако данный алгоритм тем и хорош, что не требует никаких дополнительных операций.

Его суть в том, что для всей группы вершин строится супер структура – треугольник – которая охватывает все точки. При добавлении каждой новой вершины происходит удаление тех треугольников, у которых внутрь описанных окружностей попадает данный узел. При этом образуется контур – некий многоугольник. Далее новый узел соединяется ребрами с вершинами этого контура (рис. 5).

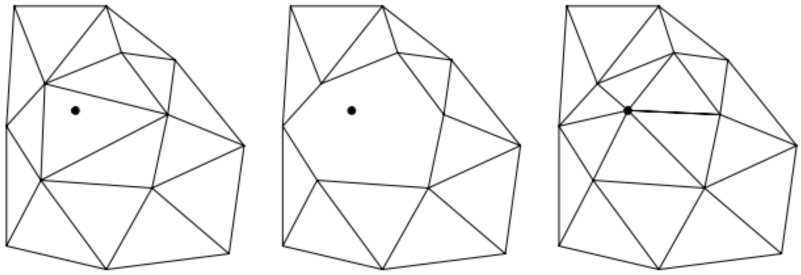

Рис. 5. Схема добавления нового узла

За счет того, что при вставке новой вершины происходит поиск треугольников, не удовлетворяющих условию Делоне, данный алгоритм считается очень удобным. Его главный недостаток – необходимо создать процедуру поиска контура многоугольника после удаления треугольников.

Литература:

1. Струц С.В., Матвеев И. А. Восстановление трехмерной структуры лица с помощью опорной модели. М., 2011.
2. Кузьмин П.В. Алгоритм реконструкции трёхмерных объектов сцены сложной формы по серии цифровых изображений. Новосибирск, 2014.
3. Скворцов А.В. Триангуляция Делоне и ее применение. Томск, 2002.