

ОГЛАВЛЕНИЕ

Вступление	2
1 Теория	3
1.1 Постановка задачи.	3
1.2 Существующие решения.	4
1.2.1 Итеративный алгоритм ближайших точек	4
1.2.2 Итеративный алгоритм ближайших точек с нормалями	4
1.3 Идеи	5
1.3.1 Симметричный итеративный алгоритм ближайших точек	5
1.3.2 Выбор исходного и целевого облаков точек	5
1.3.3 Использование инвариантов	5
2 Расстояние Хаусдорфа	7
2.1 Теоретические основы	7
2.2 Расстояние Хаусдорфа	7
2.2.1 Пример.	9
2.3 Случайные величины	10
Список литературы	11

ВСТУПЛЕНИЕ

Актуальность работы Оценка положения камеры по облакам точек лежит в основе сканирования объектов с помощью 3D сканера. Для решения этой задачи используется итеративный алгоритм ближайших точек и его модификации, однако они учитывают только расстояния между точками двух множеств. Взаимосвязь между точками одного облака содержит дополнительную информацию, которая может повысить качество оценки перемещения камеры.

Объект исследования — методы оценки параметров камеры.

Предмет исследования — алгоритмы сопоставления облаков точек.

Цель исследования. Разработка эффективного алгоритма сопоставления двух облаков точек.

Задания следующие:

- 1) исследовать существующие алгоритмы сопоставления облаков точек;
- 2) предложить альтернативный алгоритм сопоставления облаков точек.

1 ТЕОРИЯ

1.1 Постановка задачи.

Есть два облака точек: исходное S (source) и целевое T (target). К точкам исходного множества $\vec{s} \in S$ применили поворот $R \in \mathbb{R}^{3 \times 3}$ и перемещение $\vec{b} \in \mathbb{R}^3$, а также в процессе сканирования появился аддитивный гауссовый шум с неизвестной дисперсией

$$\vec{k}_s = R \cdot \vec{s} + \vec{b} + \vec{\xi}_s, \quad \vec{\xi}_s \sim N\left(\vec{0}, \sigma^2 \cdot I\right) \quad (1.1)$$

где $k : S \rightarrow T$ — функция, которая сопоставляет каждой точке из исходного множества точку из целевого множества. Воспользовавшись методом максимального правдоподобия, получаем оптимизационную задачу

$$P(k, R, \vec{b}) = \prod_{\vec{s} \in S} \frac{1}{\sqrt{2\pi\sigma^2}} \cdot e^{-\frac{\|\vec{k}_s - R \cdot \vec{s} - \vec{b}\|^2}{2\sigma^2}} \rightarrow \max_{k, R, \vec{b}}. \quad (1.2)$$

После логарифмирования, умножения на константу и отбрасывания постоянных слагаемых и множителей получаем задачу минимизации

$$E(k, R, b) = \sum_{\vec{s} \in S} \left\| \vec{k}_s - R \cdot \vec{s} - \vec{b} \right\|^2 \rightarrow \min_{k, R, b}. \quad (1.3)$$

Два множества зачастую не имеют взаимно-однозначного отображения. Это может приводить к тому, что два отрезка, расположенные под прямым углом в одном облаке точек, могут соответствовать трём отрезкам в другом облаке точек, между соседними парами которых углы по 45 градусов.

1.2 Существующие решения.

1.2.1 Итеративный алгоритм ближайших точек

Итеративный алгоритм ближайших точек (Iterative Closest Points, ICP) [1] состоит из двух чередующихся операций. Инициализируется алгоритм единичной матрицей $R = I$ и нулевым вектором смещения $\vec{b} = \vec{0}$. Первая итерация состоит в поиске разметки с фиксированной трансформацией

$$\sum_{\vec{s} \in S} \left\| \vec{k}_s - R \cdot \vec{s} - \vec{b} \right\|^2 \rightarrow \min_k. \quad (1.4)$$

На следующей итерации происходит поиск поворота и смещения при текущей разметке

$$\sum_{\vec{s} \in S} \left\| \vec{k}_s - R \cdot \vec{s} - \vec{b} \right\|^2 \rightarrow \min_{R, \vec{b}}. \quad (1.5)$$

Эти два шага повторяются, пока не будет достигнут желаемый результат, то есть пока расстояние между двумя облаками точек не будет сведено к минимуму.

1.2.2 Итеративный алгоритм ближайших точек с нормалями

Отличием данного алгоритма (Normal ICP) [2] является то, что он рассматривает каждую точку вместе с локальными особенностями поверхности

$$E(k, R, b) = \sum_{\vec{s} \in S} \alpha_{point} \cdot \left\| \vec{k}_s - R \cdot \vec{s} - \vec{b} \right\|^2 + \alpha_{plane} \cdot \left| \vec{n}_s^T \cdot (\vec{k}_s - R \cdot \vec{s} - \vec{b}) \right| \rightarrow \min_{k, R, b}, \quad (1.6)$$

где α_{point} и α_{plane} — константы, а \vec{n}_s — нормаль к точке \vec{s} на исходном облаке. Для улучшения работы алгоритма убираются

- 1) вершины, нормали которых слишком отличаются от нормалей ближайших соседей из целевого облака;
- 2) вершины, которые находятся далеко от соседей из целевого облака;
- 3) вершины, которые находятся на краю объектов.

1.3 Идеи

1.3.1 Симметричный итеративный алгоритм ближайших точек

Для каждой точки из исходного облака ищем ближайшую точку на целевом облаке. Вычисляем матрицу поворота и вектор смещения для исходного облака. Далее для каждой точки из целевого облака находим ближайшую точку на исходном облаке, оцениваем матрицу поворота и вектор смещения для целевого облака и повторяем все действия снова.

1.3.2 Выбор исходного и целевого облаков точек

В качестве исходного облака можно брать то, где точек меньше, а в качестве целевой — то, где точек больше. Таким образом нужно будет найти меньше соответствующих пар точек.

1.3.3 Использование инвариантов

Некоторые характеристики облака точек сохраняются при поворотах и перемещениях, то есть являются инвариантными относительно них. Значит, что по этим признакам можно совмещать два облака точек, не беря во вни-

мание сами трансформации, которые можно будет быстро найти при наличии разметки k . К таким свойствам относятся длины отрезков и площади замкнутых фигур [3].

2 РАССТОЯНИЕ ХАУСДОРФА

2.1 Теоретические основы

Метрическое пространство — это пара (S, d) , состоящая из множества S и метрики $d : S \times S \rightarrow \mathbb{R}$, то есть для любых $x, y, z \in S$ выполняется

- 1) $d(x, y) \geq 0$;
- 2) $d(x, y) = 0 \Leftrightarrow x = y$ — аксиома тождества;
- 3) $d(x, y) = d(y, x)$ — аксиома симметрии;
- 4) $d(x, z) \leq d(x, y) + d(y, z)$ — неравенство треугольника.

Метрическое пространство (S, d) называется сепарабельным, если существует не более чем счётное множество $\Gamma \subset S$:

$$\forall x \in S \quad \forall \varepsilon > 0 \quad \exists y \in \Gamma : \quad d(x, y) < \varepsilon.$$

Метрическое пространство (S, d) — полное, если в нём любая фундаментальная последовательность сходится.

Примером полного сепарабельного метрического пространства есть \mathbb{R}^n с евклидовым расстоянием.

2.2 Расстояние Хаусдорфа

Расстояние Хаусдорфа определяется на множестве всех непустых замкнутых подмножеств пространства \mathbb{R}^n .

Пусть E и F — непустые замкнутые подмножества \mathbb{R}^n . Расстояние Хаусдорфа между E и F определяется как

$$H(E, F) = \inf \{\varepsilon > 0 \mid E \subset F + \varepsilon, F = E + \varepsilon\}, \quad (2.1)$$

где $E + \varepsilon$ — объединение замкнутых шаров с центром в точке x радиуса ε

$$E + \varepsilon = \bigcup_{x \in E} \{\overline{B}_\varepsilon(x)\}.$$

Докажем асиомы метрики, чтобы проверить, что расстояние Хаусдорфа $H(E, F)$, заданое формулой 2.1 — это действительно расстояние между двумя замкнутыми множествами в \mathbb{R}^n .

- 1) $H(E, F) \geq 0$. Это следует из определения 2.1, так как точная нижняя грань величины $\varepsilon > 0$ неотрицательна.
- 2) $H(E, F) = 0$ тогда и только тогда, когда $E = F$. Если $E = F$, то $E \subset F$ и $F \subset E$, то есть $H(E, F) = \varepsilon = 0$. С другой стороны, если $H(E, F) = 0$, то $\varepsilon = 0$, то есть опять $E \subset F$ и $F \subset E$. Получаем, что $E = F$.
- 3) $H(E, F) \leq H(E, F) + H(F, G)$ для любых замкнутых множеств E, F, G из \mathbb{R}^n . Запишем правую часть по определению 2.1

$$\begin{aligned} H(E, F) + H(F, G) &= \\ &= \inf \{\varepsilon_1 > 0 : E + \varepsilon_1 \subset F, F + \varepsilon_1 \subset E\} + \\ &\quad + \inf \{\varepsilon_2 > 0 : F + \varepsilon_2 \subset G, G + \varepsilon_2 \subset F\}. \end{aligned}$$

Сумма инфимумов равна инфимуму суммы. Объединяя два множества, получаем

$$E + \varepsilon_1 \subset F \subset F + \varepsilon_2 \subset G, G + \varepsilon_2 \subset F \subset F + \varepsilon_1 \subset E.$$

Собирая первые и последние части выражений, видим, что $E + \varepsilon \subset G$ и $G + \varepsilon \subset E$, а это левая часть неравенства.

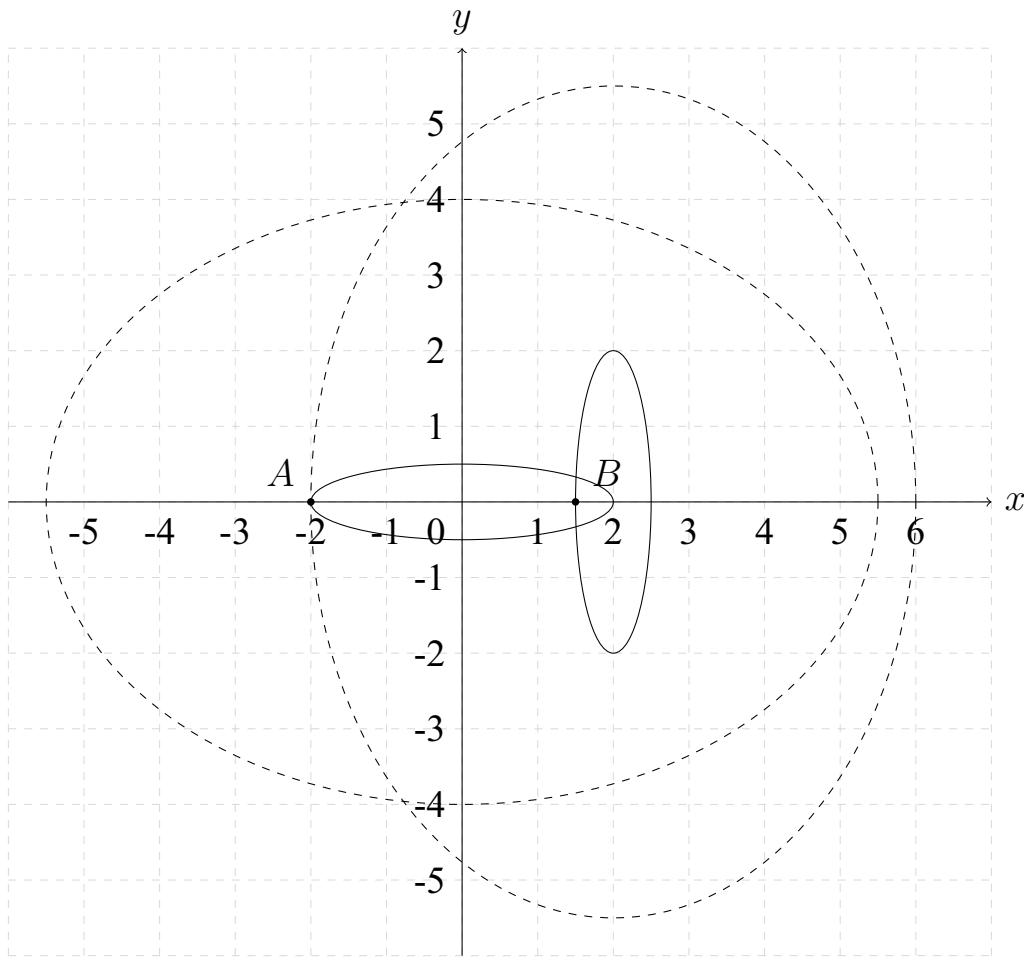


Рисунок 2.1 – Эллипсы, между которыми ищем расстояние Хаусдорфа

2.2.1 Пример

Найдём расстояние Хаусдорфа между двумя эллипсами 2.1 [4]

$$E : \frac{x^2}{4} + 4y^2 = 1,$$

$$F : 4(x - 2)^2 + \frac{y^2}{4} = 1.$$

Пунктиром нарисованы эллипсы $E + \varepsilon$ и $F + \varepsilon$ такие, чтобы выполнялось 2.1. Таким образом ε в данном случае — это расстояние между точками A и B , которое равно $\varepsilon = 1.5 - (-2) = 3.5$. Поэтому $H(E, D) = 3.5$.

2.3 Случайные величины

Имеем вероятностное пространство $(\Omega, \mathcal{F}, \mathbb{P})$, где $\Omega = \mathbb{R}^n$, $\mathcal{F} = \mathcal{B}(\mathbb{R}^n)$ – борелевская σ -алгебра на \mathbb{R}^n , \mathbb{P} – вероятностная мера на \mathcal{F} .

Есть два определения случайной величины:

- 1) функция $\xi : \Omega \rightarrow \mathbb{R}$ называется случайной величиной, если

$$\forall \Delta \in \mathcal{B}(\mathbb{R}) : \quad \xi^{-1}(\Delta) = \{\omega : \quad \xi(\omega) \in \Delta\} \in \mathcal{F};$$

- 2) функция $\xi : \Omega \rightarrow \mathbb{R}$ называется случайной величиной, если

$$\forall c \in \mathbb{R} : \quad \{\omega : \quad \xi(\omega) \leq c\} = \xi^{-1}((-\infty, c]) \in \mathcal{F}.$$

Хотим проверить, является ли расстояние Хаусдорфа

$$\gamma = H\left(AE + \vec{b} + \vec{\xi}, F\right) = H(\xi(E), F)$$

случайной величиной, где ξ – случайное отображение из \mathbb{R}^n в \mathbb{R}^n .

По определению 2.1 $\gamma = \inf_n \{\varepsilon_n : \quad \xi(E) + \varepsilon_n \subset F, F + \varepsilon_n \subset \xi(E)\}$.

Чтобы это проверить, нужно выяснить, является ли случайной величиной число $\varepsilon_n > 0$. Если получится, то

$$\forall c \in \mathbb{R} : \quad \left\{ \omega : \quad \inf_n \varepsilon_n(\omega) \leq c \right\} = \bigcup_{n=1}^{\infty} \left\{ \omega : \quad \varepsilon_n(\omega) \leq c \right\}.$$

Каждое множество, которые стоит под знаком объединения, по определению, принадлежит σ -алгебре \mathcal{F} , и их счётное объединение принадлежит σ -алгебре \mathcal{F} , так как она замкнута относительно счётной операции объединения.

СПИСОК ЛИТЕРАТУРЫ

- 1 Zhang, Zhengyou. Iterative point matching for registration of free-form curves and surfaces / Zhengyou Zhang // *International journal of computer vision*. — 1994. — October. — Vol. 13. — Pp. 119–152.
- 2 Li, Hao. Global Correspondence Optimization for Non-Rigid Registration of Depth Scans / Hao Li, Robert W. Sumner, Mark Pauly // *Computer Graphics Forum (Proc. SGP'08)*. — 2008. — July. — Vol. 27, no. 5.
- 3 Hartley, R. Multiple View Geometry in Computer Vision / R. Hartley, A. Zisserman. — Cambridge University Press, 2004.
- 4 Crownover, R.M. Introduction to fractals and chaos / R.M. Crownover // Jones and Bartlett books in mathematics. — Jones and Bartlett, 1995.