

Модели коррекции кватернионов ориентации в инерциальной навигации

Гадаев Тамаз

Московский физико-технический институт
Физтех-школа прикладной математики и информатики
Кафедра интеллектуальных систем

Научный руководитель д.ф.-м.н. В. В. Стрижов
Москва,
2021 г.



Задача инерциальной навигации с неточными датчиками

Цель

Построить алгоритм восстановления траектории движения пешехода с мобильным устройством с помощью модели его скорости по данным с инерциальных датчиков: акселерометр и гироскоп.

Проблема

Инерциальные датчики в мобильных устройствах имеют высокий уровень шума, что делает невозможным решение задачи навигации при помощи физической модели.

Метод

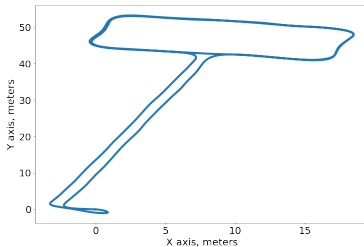
Нейросетевая модель в сочетании с физическими предположениями о данных, коррекцией ориентации мобильного устройства и использование пополнений исходной выборки.

Восстановление траектории движения пешехода

Три временных ряда с одинаковой временной сеткой: данные акселерометра \mathbf{a} , данные гироскопа \mathbf{w} и координаты человека \mathbf{r} с устройством, содержащим датчики

$$\mathbf{e} = \{\mathbf{a}, \mathbf{w}, \mathbf{r}\}, \quad \mathbf{a}, \mathbf{w}, \mathbf{r} \in \mathbb{R}^{T \times 3}.$$

Выборка — набор экспериментов $\mathcal{D} = \{e^i\}_{i=1}^n$, разделенный на непересекающиеся обучающую и тестовую части $\mathcal{D} = \mathcal{D}_T \cup \mathcal{D}_L$



Пример данных: слева — данные с датчиков, справа — траектория движения

Постановка задачи

Задача: Оценить траекторию движения человека $\hat{\mathbf{r}}$ по данным с датчиков (\mathbf{a}, \mathbf{w}) .

На отрезках длины Δt временных рядов с датчиков обучается модель скорости человека

$$M(\mathbf{a}_{t:t+\Delta t}^i, \mathbf{w}_{t:t+\Delta t}) : \mathbb{R}^{\Delta t \times 3} \times \mathbb{R}^{\Delta t \times 3} \rightarrow \mathbb{R}^3$$
$$W = \sum_{e^i \in \mathcal{D}_L} \sum_{t \in \tau} (M(\mathbf{a}_{t:t+\Delta t}^i, \mathbf{w}_{t:t+\Delta t}^i) - \overline{\dot{\mathbf{r}}}_{t:t+\Delta t}^i)^2 \rightarrow \min$$

Для получения оценки траектории $\hat{\mathbf{r}}_{t:t+\Delta t}$ оценки средних скоростей за интервал численно интегрируются и приводятся к изначальной временной сетке с помощью интерполяции.

Модель оценивается по среднеквадратичному отклонению траектории RMSE и относительной ошибке позиции MPE:

$$RMSE(\hat{\mathbf{r}}, \mathbf{r}) = \sqrt{\frac{1}{T} \sum_{t=1}^T |\mathbf{r}_t - \hat{\mathbf{r}}_t|^2}.$$

$$MPE(\hat{\mathbf{r}}, \mathbf{r}) = \frac{1}{T} \sum_{t=1}^T \frac{|\mathbf{r}_t - \hat{\mathbf{r}}_t|}{|\mathbf{r}_t|}.$$

Предположение: точность модели выше, если данные с датчиков и скорости приведены в одну систему координат

- Векторы ускорения \mathbf{a}^I и поворота \mathbf{w}^I с датчиков снимаются в собственной системе координат I , которая вращается и ускоряется вместе с устройством.
- Траектория, из которой вычисляется скорость человека, дана в мировой системе координат W .
- Для перехода из I и W используются данные гироскопа.

Переход из I в W

- Кватернион q_t^{rot} задаёт вращение устройства в промежуток времени от t до $t + 1$.
- Кватернион ориентации устройства $q_t^{WI} = \prod_{i=0}^t q_{t-i}^{rot}$ задаёт поворот, приводящий систему координат I к системе W в момент времени t .
- Кватернион $q_t^{IW} = q_t^{WI*}$ задаёт поворот приводящий систему координат W к системе I в момент времени t .

Стабилизированная система координат

Чтобы исключить влияние поворотов на углы крена и тангажа, данные с датчиков и скорости переведены в стабилизированную систему координат S , связанную с устройством, в которой ось z направлена вдоль вектора гравитации.

Поворот приводящий систему I к системе S задаётся как минимальный поворот от единичного вектора вдоль оси z к вектору гравитации \mathbf{g}_t^I кватернионом $q_t^{SI} = \text{MinRot}(\mathbf{g}_t^I, \mathbf{e}_z)$.

Для перевода в S необходимо вычислить вектор гравитации \mathbf{g}_t^I в каждый момент времени t в системе I .

Предположение:

$$\mathbf{a}^W = \mathbf{a}_{device}^W + |\mathbf{g}| \mathbf{e}_z^W.$$

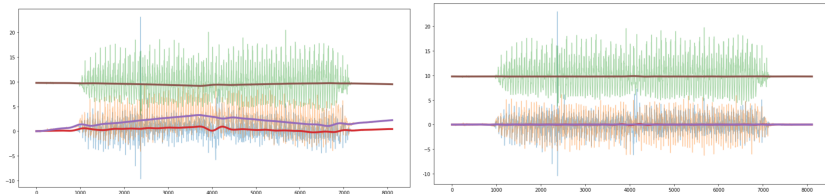
Аппроксимируем $\hat{\mathbf{g}}^W = \text{LP}(\mathbf{a}^W)$, где LP — низкочастотный фильтр.

Тогда $\hat{\mathbf{g}}_t^I = q_t^{IW}(\hat{\mathbf{g}}_t^W)$, $q_t^{SI} = \text{MinRot}(\text{LP}(q_t^{WI}(\mathbf{a}_t^I), \mathbf{e}_z)$

Коррекция кватернионов ориентации

Скорости $\dot{\mathbf{r}}^S = q^{SI}(q^{IW}(\dot{\mathbf{r}}^W))$ получаются с использованием q^{WI} , которые строятся по данным гироскопа. Эти данные неточные и накапливают ошибку.

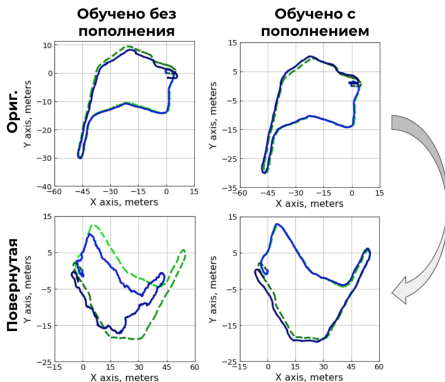
Для коррекции q^{WI} используется предположение $\mathbf{a}^W = \mathbf{a}_{device}^W + |\mathbf{g}|\mathbf{e}_z^W$. Наименьший поворот между $LP(\mathbf{a}^W) = LP(q^{WI}(\mathbf{a}^I))$ и \mathbf{e}_z^W указывает на дрейф в гироскопе. Введение $q_t^{corr} = \text{MinRot}(LP(q^{WI}(\mathbf{a}^I)), \mathbf{e}_z^W)$ позволяет скорректировать эту ошибку.



Слева $LP(\mathbf{a}^W)$ до коррекции, справа — после

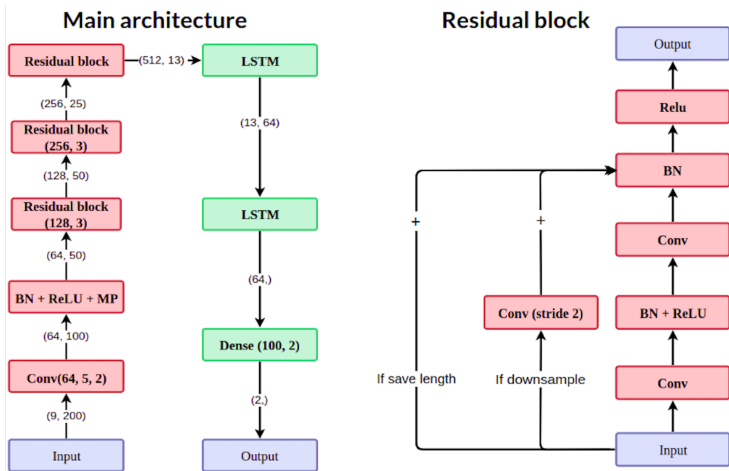
Пополнение выборки

Чтобы сделать модель более устойчивой, к отрезкам временных рядов и скоростям были применены повороты на случайный угол в плоскости движения.



Модель, обученная на данных со случайными поворотами, демонстрирует большую устойчивость к повороту траектории и данных с сенсоров.

Нейросетевая модель



Архитектура нейросетевой модели

Итоговый алгоритм восстановления траектории

Системы координат:

- 1 Получение данных с датчиков на отрезках
- 2 Вычисление вектора гравитации g_t^I в I системе в каждый момент времени t
- 3 Получение кватернионов q^{SI}
- 4 Получение кватернионов q^{WI}
- 5 Коррекция кватернионов q^{WI}
- 6 Перевод данных в S систему
- 7 Предсказание скорости по данным
- 8 Перевод скорости в W
- 9 Интегрирование скоростей

Цель:

- Сравнить предложенный алгоритм с аналогом
- Проанализировать вклад каждой компоненты алгоритма в его точность

Выборка:

Для обучения и тестирования модели взята выборка RuDaCOP. В каждом эксперименте пешеход имел при себе устройство с инерциальными датчиками, расположенное в разных местах: карман брюк, рука, сумка, нагрудный карман и т. п.

Сравнение точности алгоритмов

Модель	RMSE	MPE
RoNIN	13.0	11.2
RNL	12.1	10.1
RNL + OC	10.9	8.9
RNL + OC + Aug	10.5	8.5

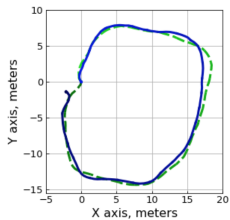
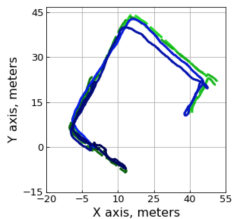
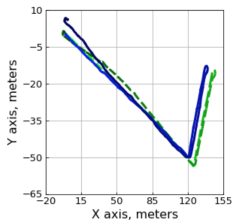
RoNIN — аналогичный существующий подход к задаче инерциальной навигации на основе нейросетей

RNL — использование архитектуры ResNetLSTM

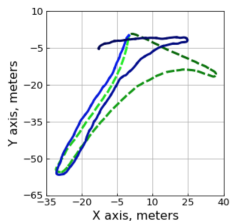
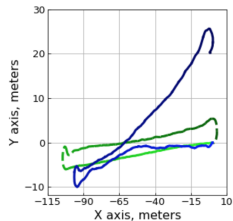
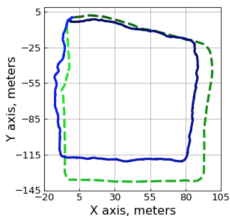
OC — использование коррекции кватернионов ориентации устройства с помощью вектора гравитации

Aug — пополнения выборки

Анализ ошибки: примеры



Примеры адекватных предсказаний модели



Примеры ошибок

- 1 Решена задача инерциальной навигации в терминах обучения модели восстановления скорости по инерциальным датчикам
- 2 Предложен алгоритм вычисления вектора гравитации по инерциальным датчикам
- 3 Предложен алгоритм коррекции кватернионов ориентации устройства с помощью вектора гравитации
- 4 Исследованы ошибки полученной модели