Лекция 12. Практические примеры решения задач. Анализ поведения животных.

А. С. Конушин[1] Д. П. Ветров[2] Д. А. Кропотов[3]
В. С. Конушин[1] О. В. Баринова[1]

[1]МГУ, ВМиК, лаб. КГ

[2]МГУ, ВМиК, каф. ММП

[3]ВЦ РАН

Спецкурс «Структурные методы анализа изображений и сигналов»

План

Трекинг одной мыши в клетке

Множественный трекинг мышей

Сегментация поведения на поведенческие акты

Структурный анализ поведения

План

Трекинг одной мыши в клетке

Множественный трекинг мышей

Сегментация поведения на поведенческие акты

Структурный анализ поведения

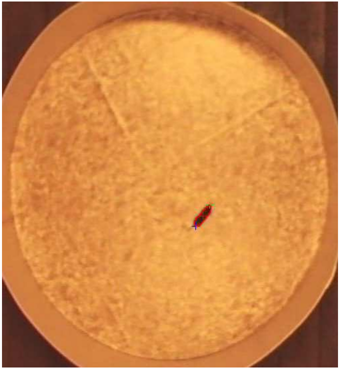
Задача трекинга

Необходимо по видеозаписи поведения восстановить для каждого кадра:

- Контур мыши
- Характерные точки: нос, хвост, центр масс

Определение контура мыши

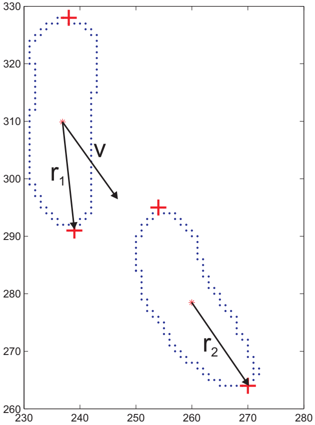
$\Rightarrow$



Задача выделения контура мыши может быть решена с помощью разрезов графов, но есть способ проще:

- Определение фона: $\text{Background} = \text{Median}(\text{Image}_1, \ldots, \text{Image}_n)$
- Отсечение по порогу: $\text{Mouse} = \text{Image}(\text{Image} < \alpha \cdot \text{Background})$

Определение точек носа/хвоста



Пусть для каждого контура определены две точки, соответствующие носу и хвосту (две самые удаленные точки от центра). $x_i = 0$, если нос – первая точка, $x_i = 1$, если нос – вторая точка.

Оптимизационная задача:

$$Q = f(x_1) + \sum_{i=2}^{N} f(x_i, x_{i-1}) \to \min_{\{x_i\}}$$

$$f(x_i, x_{i-1}) = \angle(\boldsymbol{r_1}, \boldsymbol{r_2}) + \lambda \frac{\langle \boldsymbol{r}_2, \boldsymbol{v} \rangle}{\|\boldsymbol{v}\|^2}$$

Определение точек носа/хвоста

Оптимизационная задача

$$Q = f(x_1) + \sum_{i=2}^{N} f(x_i, x_{i-1}) \to \min_{\{x_i\}}$$

Такая задача решается динамическим программированием.
Проход вперед:

$$V_1(x_1) = f(x_1);$$
$$V_i(x_i) = \min_{x_{i-1}} [V_{i-1}(x_{i-1}) + f(x_i, x_{i-1})]$$
$$S_i(x_i) = \arg\min_{x_{i-1}} [V_{i-1}(x_{i-1}) + f(x_i, x_{i-1})]$$

Проход назад:

$$x_N^* = \arg\min_{x_i} V_N(x_i)$$
$$x_i^* = S_{i+1}(x_{i+1}^*)$$

План

Трекинг одной мыши в клетке

Множественный трекинг мышей

Сегментация поведения на поведенческие акты

Структурный анализ поведения

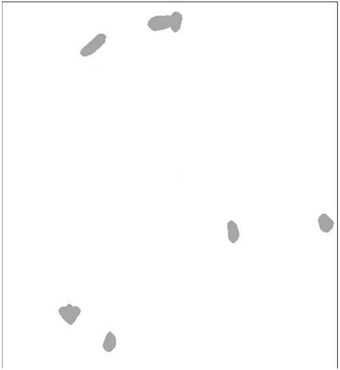
Задача множественного трекинга

Особенности задачи:

- Возможность полных перекрытий животных
- Необходимость определения дополнительно к форме животных положений точек носа и хвоста
- Необходимость использования простейших моделей движения мыши
- Необходимость обработки длительных видео

Вычитание фона

$\Rightarrow$

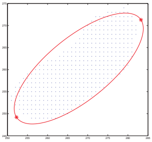


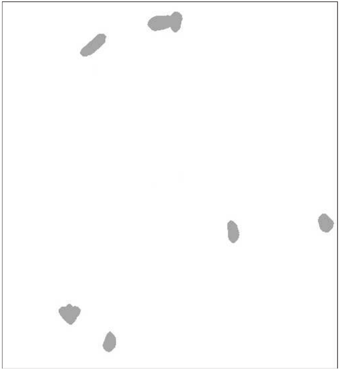
$$\text{Background} = \text{Median}(\text{Image}_1, \ldots, \text{Image}_n)$$
$$\text{Mice} = \text{Image}(\text{Image} < \alpha \cdot \text{Background})$$

Входная информация для алгоритма множественного трекинга – набор пятен для каждого кадра

Модель мыши

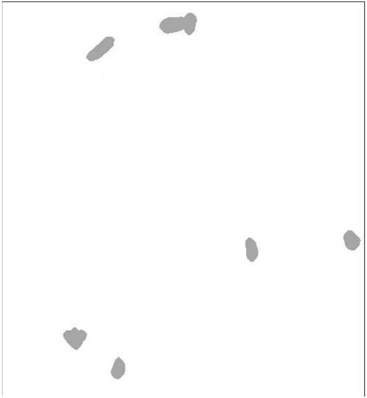
Модель мыши — эллипс



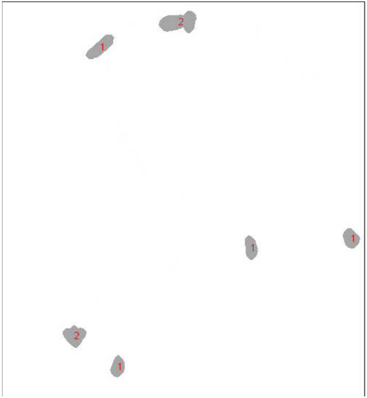


$\Rightarrow$

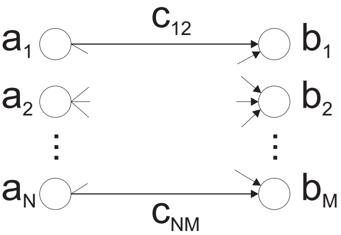
Определение количества мышей в блобах



$\Rightarrow$



Транспортная задача (задача о назначении)



$$\sum_{i,j=1}^{N,M} c_{ij}x_{ij} \to \min_{\{x_{ij}\}}$$

$$\sum_{j=1}^{M} x_{ij} = a_i,\ \ i = \overline{1,N}$$

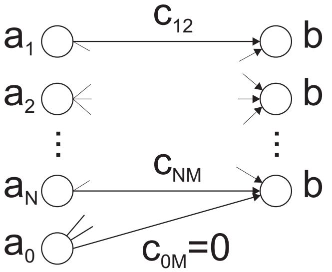
$$\sum_{i=1}^{N} x_{ij} = b_j,\ \ j = \overline{1,M}$$

$$x_{ij} \geq 0$$

При целочисленных a_i и b_j решение x_{ij} также получается целочисленным.

Определение количества мышей в блобах

Предполагается, что в первый момент времени все мыши разделены



$$\sum_{i=0,j=1}^{N,M} c_{ij}x_{ij} \to \min_{\{x_{ij}\}}$$

$$\sum_{j=1}^{M} x_{ij} = a_i,\ \ i = \overline{1,N}$$

$$\sum_{i=0}^{N} x_{ij} = b,\ \ j = \overline{1,M}$$

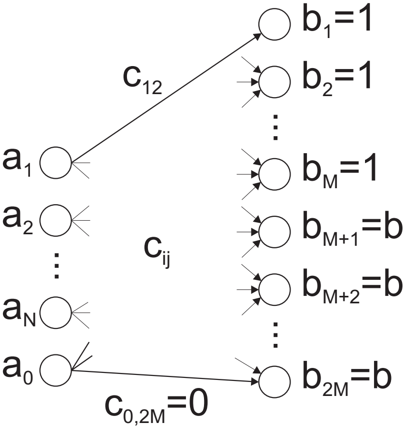
$$x_{ij} \geq 0$$

$$b = \sum_{i=1}^{N} a_i, a_0 = b(M-1)$$

$$c_{0j} = 0,\ \ j = \overline{1,M}$$

a_i — число мышей в i-ом блобе в предыдущий момент времени
b — максимально возможное число мышей в блобе в текущий момент времени

Определение количества мышей в блобах



$$\sum_{i=0,j=1}^{N,2M} c_{ij}x_{ij} \to \min_{\{x_{ij}\}}$$

$$\sum_{j=1}^{2M} x_{ij} = a_i,\ i = \overline{1,N}$$

$$\sum_{i=0}^{N} x_{ij} = b_j,\ j = \overline{1,2M}$$

$$x_{ij} \geq 0$$

$$b_j = 1,\ c_{0j} = +\inf,\ j = \overline{1,M}$$

$$b_j = \sum_{i=1}^{N} a_i - M,\ j = \overline{M+1,2M}$$

$$c_{0j} = 0,\ j = \overline{M+1,2M}$$

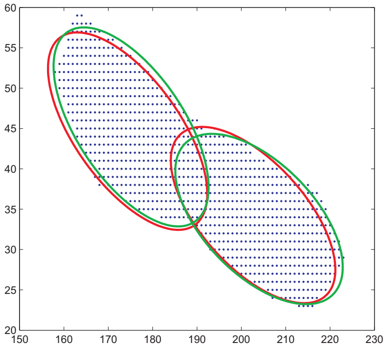
$$a_0 = (M-1)(\sum_{i=1}^{N} a_i - M)$$

Процедура разделения блобов

Оптимизационная задача:

$$p(\boldsymbol{e}_1, \boldsymbol{e}_2 | X) \propto p(X | \boldsymbol{e}_1, \boldsymbol{e}_2) p(\boldsymbol{e}_1) p(\boldsymbol{e}_2) \to \max_{\boldsymbol{e}_1, \boldsymbol{e}_2}$$

X — точки блоба, $\boldsymbol{e}_1, \boldsymbol{e}_2$ — параметры эллипсов



EM-алгоритм для разделения смеси распределений

$$p(\boldsymbol{x}) = \sum_{k=1}^{M} w_k \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_k, \Sigma_k)$$

Е-шаг:

$$\gamma(z_{ik}) = \frac{w_k \mathcal{N}(\boldsymbol{x}_i|\boldsymbol{\mu}_k, \Sigma_k)}{\sum_{j=1}^{M} w_j \mathcal{N}(\boldsymbol{x}_i|\boldsymbol{\mu}_j, \Sigma_j)}$$

М-шаг:

$$w_k = \frac{1}{N} \sum_{i=1}^{N} \gamma(z_{ik})$$

$$\boldsymbol{\mu}_k = \frac{1}{w_k N} \sum_{i=1}^{N} \gamma(z_{ik}) \boldsymbol{x}_i$$

$$\Sigma_k = \frac{1}{w_k N} \sum_{i=1}^{N} \gamma(z_{ik}) (\boldsymbol{x}_i - \boldsymbol{\mu}_k)(\boldsymbol{x}_i - \boldsymbol{\mu}_k)^T$$

EM-алгоритм. E-шаг.

$X = \{\boldsymbol{x}_i\}$ — точки блоба
C_k — k-ый эллипс, $\boldsymbol{m}_k$ — его центр

$$\gamma(z_{ik}) = \begin{cases} 1, & \boldsymbol{x}_i \in C_k \text{ или } k = \arg\min\limits_j \rho_M(\boldsymbol{x}_i, \boldsymbol{m}_j) \\ 0, & \text{иначе} \end{cases}$$

EM-алгоритм. M-шаг.

Пусть α — угол наклона наибольшей полуоси эллипса на предыдущем кадре

$\boldsymbol{y}_i$ — центрированные и повернутые на угол α точки блоба

Функция правдоподобия:

$$p(Y|A_k) = (\det A_k)^{n_k} \exp\left(-\frac{1}{2}\sum_i \boldsymbol{y}_i^T A_k \boldsymbol{y}_i\right)$$

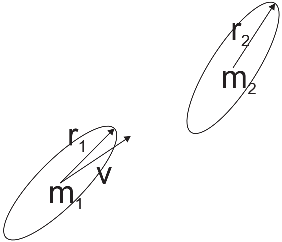
Априорное распределение:

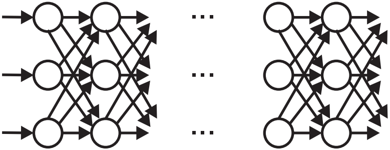
$$p(A_k|V,\nu) = \text{Const}\,(\det A_k)^{(\nu-3)/2} \exp\left(-\frac{1}{2}\operatorname{tr}(V^{-1}A_k)\right)$$

В силу сопряженности точка максимума апостериорной плотности выписывается анатилически:

$$A_k = (n_k + \nu - 3)\left(\sum_i \gamma(z_{ik})\boldsymbol{y}_i\boldsymbol{y}_i^T + V^{-1}\right)^{-1}$$

Динамическое программирование для множественного трекинга





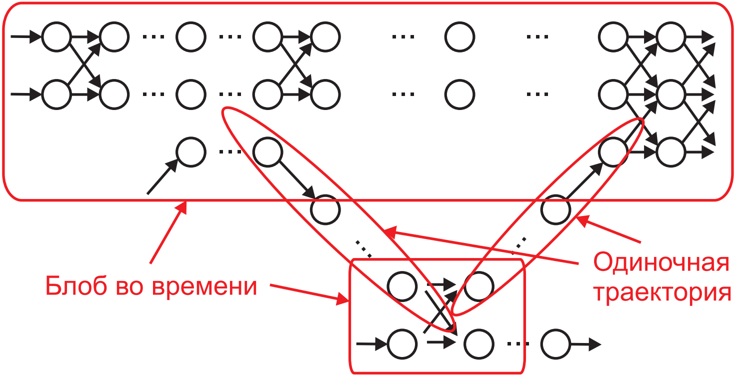
$x_i = (n_i, p_i)$, $n_i \in \{1, \dots, K\}$ — номер мыши, $p_i \in \{0, 1\}$ — индикатор точки носа. Размер функции Беллмана $K!2^K$.

Оптимизационная задача:

$$Q = f(x_1) + \sum_{i=2}^{N} f(x_i, x_{i-1}) \to \min_{\{x_i\}}$$

$$f(x_i, x_{i-1}) = \angle(\boldsymbol{r_1}, \boldsymbol{r_2}) + \lambda_1 \frac{\langle \boldsymbol{r}_2, \boldsymbol{v} \rangle}{\|\boldsymbol{v}\|^2} + \lambda_2 \rho(\boldsymbol{m}_1, \boldsymbol{m}_2)$$

Структура данных, блобы во времени



Предположение: в каждый момент времени в группу мышей могут либо добавляться либо отниматься одиночные мыши

Этапы алгоритма

- Определение количества мышей в каждом блобе
- Определение точек носа/хвоста для траекторий одиночных мышей
- Разделение блобов на набор эллипсов
- Идентификация и определение точек носа/хвоста внутри блобов во времени
- Сшивка траекторий мышей
- Окончательное определение точек носа/хвоста для полных траекторий

Сделанные предположения

- Все мыши на первом кадре разделены
- Мыши хорошо описываются эллипсами
- При слипании мышей в каждый момент времени происходит только прибавление мышей к общей группе или удаление мышей из общей группы, причем добавляются или удаляются только одиночные мыши

План

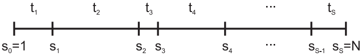
Трекинг одной мыши в клетке

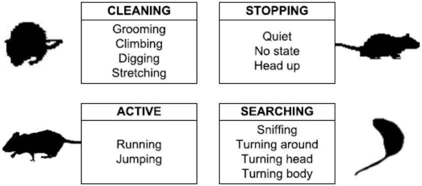
Множественный трекинг мышей

Сегментация поведения на поведенческие акты

Структурный анализ поведения

Постановка задачи



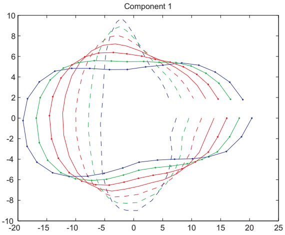


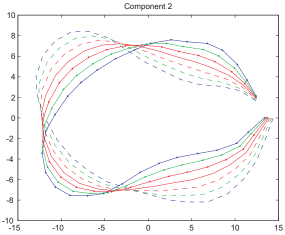
По наблюдаемым данным X разбить сигнал на набор сегментов $[s_i + 1,\ s_{i+1}]$, соответствующих отдельным поведенческим актам $t_i \in \{\mathrm{Grooming}, \mathrm{Climbing}, \ldots, \mathrm{Running}\}$.
Для решения используем скрытые марковские модели.

Признаковое пространство

Признаки, вычисляемые по данным видеотрекинга:

- Величина скорости
- Величина ускорения
- Угол между скоростью и ускорением
- Расстояние до края/центра арены
- Проекции на собственные контура
- И т.д.



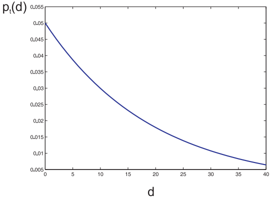


Априорное распределение на длину одного сегмента

В классической СММ априорная вероятность нахождения в состоянии t ровно d моментов времени составляет

$$p_t(d) = (A_{tt})^d(1 - A_{tt})$$

Здесь A_{tt} — вероятность остаться в состоянии t в каждый момент времени.



Поведенческие акты не могут быть очень короткими, поэтому более адекватной моделью будет сдвинутое геометрическое распределение.

План

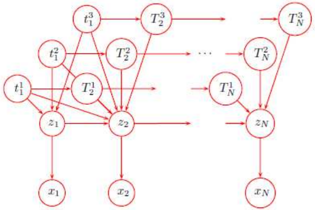
Трекинг одной мыши в клетке

Множественный трекинг мышей

Сегментация поведения на поведенческие акты

Структурный анализ поведения

Композитно-иерархическая СММ



Здесь $t_i \in \{0, 1\}$ — факт активности i-ого эмоционального контура, $z_i \in \{1, \dots, K\}$ — поведенческие акты, ограничения композитности:

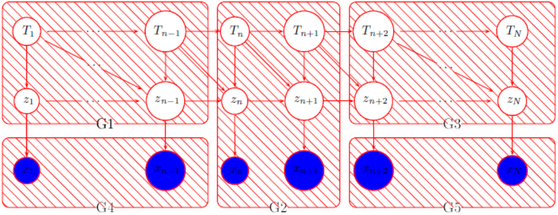
$$\sigma = \tau \Rightarrow p(z_n = \zeta | z_{n-1} = \eta, T_n = \sigma, T_{n-1} = \tau) =$$
$$= \frac{1}{|N(\sigma)|} \sum_{i \in N(\sigma)} p(z_n = \zeta | z_{n-1} = \eta, T_n = \sigma_i, T_{n-1} = \tau_i)$$

$$\sigma \neq \tau \Rightarrow p(z_n = \zeta | z_{n-1} = \eta, T_n = \sigma, T_{n-1} = \tau) = p(z_1 = \zeta, T_1 = \sigma) =$$
$$= \frac{1}{|N(\sigma)|} \sum_{i \in N(\sigma)} p(z_1 = \zeta, T_1 = \sigma_i)$$

Задачи анализа композитно-иерархической СММ

Сегментация. Решается прямой модификацией алгоритма Витерби.

Обучение без учителя. Е-шаг ЕМ-алгоритма на основе алгоритма sum-product.



На М-шаге необходимо учесть ограничения композитности при пересчете $\boldsymbol{w}$ и A. Можно реализовать с использованием итерационного процесса.