Архитектура MediaPipe Hands

Двухэтапный подход

Система использует конвейерную обработку:

```
Изображение → Детектор ладони → Ключевые точки → Фильтрация → Действие
HandLandmark.WRIST              # 0 - запястье
HandLandmark.THUMB_TIP          # 4 - кончик большого пальца
HandLandmark.INDEX_FINGER_TIP   # 8 - кончик указательного пальца
HandLandmark.MIDDLE_FINGER_TIP  # 12 - кончик среднего пальца
HandLandmark.PINKY_TIP          # 20 - кончик мизинца
HandLandmark.MIDDLE_FINGER_MCP  # 9 - основание среднего пальца
21 нейрон
```

Сверточные нейронные сети (CNN)

Основная операция свертки, используемая в MediaPipe:

$$(F * I)[x, y] = \sum_{i=-k}^{k} \sum_{j=-k}^{k} F[i, j] \cdot I[x + i, y + j]$$

где F - фильтр ядра, I - входное изображение.

Эта оптимизация снижает вычислительную сложность с $O(k^2 \cdot C^2)$ до $O(k^2 \cdot C + C^2)$.

Математические модели жестов

Евклидово расстояние для определения жестов

Расстояние между ключевыми точками используется для распознавания жестов:

$$d(i, j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2}$$

Пример применения:

- ЛКМ: $d(\text{index\_tip}, \text{thumb\_tip}) > \text{threshold}$
- ПКМ: $d(\text{pinky\_tip}, \text{fixed\_point}) > \text{threshold}$

Угловые характеристики для скроллинга

Косинус угла между вектором пальца и кисти:

$$\cos\ \theta = \frac{\mathbf{v} \cdot \mathbf{w}}{\| \mathbf{v} \| \cdot \| \mathbf{w} \|}$$

где:

- $\mathbf{v}$ - вектор от сустава к кончику пальца
- $\mathbf{w}$ - вектор от запястья к суставу

АЛГОРИТМЫ ФИЛЬТРАЦИИ И СГЛАЖИВАНИЯ

Экспоненциальное сглаживание

Для плавного перемещения курсора используется рекуррентное сглаживание:

$$S_t = \alpha x_t + (1 - \alpha) S_{t-1}$$

где:

- S_t - сглаженное значение в момент t
- x_t - новое измерение
- $\alpha \in [0,1]$ - коэффициент сглаживания (в проекте $\alpha = 0.85$)

Кальмановский фильтр (упрощенная версия)

Для предсказания движения курсора:

Уравнение состояния:

$$x_t = x_{t-1} + v_{t-1}\Delta t + w_t$$

Уравнение измерения:

$$z_t = x_t + v_t$$

где $w_t \sim \mathcal{N}(0, Q)$ и $v_t \sim \mathcal{N}(0, R)$ - шумы процесса и измерения.

Медианная фильтрация для устранения выбросов

$$y_t = \text{median}\{x_{t-2}, x_{t-1}, x_t, x_{t+1}, x_{t+2}\}$$

ПРЕОБРАЗОВАНИЕ КООРДИНАТ

Линейное отображение координат

Из системы координат камеры в экранные координаты:

$$\begin{aligned} x_{\text{screen}} &= \frac{x_{\text{camera}} - x_{\min}}{x_{\max} - x_{\min}} \cdot W_{\text{screen}} \\ y_{\text{screen}} &= \frac{y_{\text{camera}} - y_{\min}}{y_{\max} - y_{\min}} \cdot H_{\text{screen}} \end{aligned}$$

Модель с инерцией

Для естественного движения курсора:

$$\begin{aligned} \mathbf{v}_t &= \gamma \mathbf{v}_{t-1} + (1-\gamma)\frac{\mathbf{p}_t - \mathbf{p}_{t-1}}{\Delta t} \\ \mathbf{p}_t^{\text{pred}} &= \mathbf{p}_{t-1} + \mathbf{v}_t \Delta t \end{aligned}$$

где $\gamma = 0.1$ - коэффициент инерции.

ВЕРОЯТНОСТНЫЕ МОДЕЛИ

Байесовская классификация жестов

$$P(G \mid F) = \frac{P(F \mid G) \cdot P(G)}{P(F)}$$

где:

- $P(G \mid F)$ - вероятность жеста G при признаках F
- $P(F \mid G)$ - правдоподобие признаков
- $P(G)$ - априорная вероятность жеста

Пороговая логика с гистерезисом

```
text
```

```
Если расстояние > threshold_high → жест активен
Если расстояние < threshold_low → жест неактивен
Иначе → сохранить предыдущее состояние
```

МЕТРИКИ ОЦЕНКИ КАЧЕСТВА

Точность и полнота

$$\begin{aligned} \text{Precision} &= \frac{TP}{TP + FP} \\ \text{Recall} &= \frac{TP}{TP + FN} \\ F_1 &= 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \end{aligned}$$

PCK (Percentage of Correct Keypoints)

$$\text{PCK@}\alpha = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}\left(\frac{\| \hat{p}_i - p_i \|_2}{d_{\text{ref}}} < \alpha\right)$$

где $\alpha = 0.2$ - порог точности.

Временные метрики

- **FPS (Frames Per Second)**: $\text{FPS} = \frac{N_{\text{frames}}}{\Delta T}$
- **Задержка**: $\tau = t_{\text{output}} - t_{\text{input}}$
- **Jitter**: $J = \sqrt{\frac{1}{N}\sum(\tau_i - \bar{\tau})^2}$

ОПТИМИЗАЦИЯ ПРОИЗВОДИТЕЛЬНОСТИ

Адаптивное управление качеством

$$\text{Quality}_{t+1} = \begin{cases} \text{Quality}_t \cdot 0.9 & \text{если FPS} < 30 \\ \min(1.0, \text{Quality}_t \cdot 1.1) & \text{если FPS} > 40 \end{cases}$$

Квантование модели

Преобразование весов из FP32 в INT8:

$$w_{\text{int8}} = \text{round}\left(\frac{w_{\text{float}} - \min}{\max - \min} \cdot 255\right)$$

Ускорение в 3-4 раза при потере точности <1%.

ЭКСПЕРИМЕНТАЛЬНЫЕ РЕЗУЛЬТАТЫ

Ключевые формулы системы

Основные математические модели:

1. **Распознавание жестов**:

$$d(i,j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

2. **Сглаживание движения**:

$$S_t = \alpha x_t + (1 - \alpha) S_{t-1}$$

3. **Преобразование координат**:

$$x_{\text{screen}} = \frac{x_{\text{camera}} - x_{\min}}{x_{\max} - x_{\min}} \cdot W_{\text{screen}}$$

4. **Оценка качества**:

$$F_1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

Система демонстрирует эффективное сочетание современных нейросетевых технологий с классическими алгоритмами обработки сигналов, обеспечивая естественный и интуитивный интерфейс для бесконтактного управления компьютером.

def calculate_palm_center(self, landmarks):

Математическая основа:

$$C_x = \frac{x_{\text{wrist}} + x_{\text{middle\_mcp}}}{2}, C_y = \frac{y_{\text{wrist}} + y_{\text{middle\_mcp}}}{2}$$

где C - центр ладони, используемый как реперная точка для всех жестов.

def get_finger_tip_distance(self, finger_tip, palm_center):

Математическая модель:

$$d = \sqrt{(x_{\text{finger}} - C_x)^2 + (y_{\text{finger}} - C_y)^2}$$

где d - нормализованное расстояние (0-1), используемое для определения состояния пальца.

def is_finger_raised(self, distance):

Пороговая логика:

$$\text{Состояние} = \begin{cases} \text{raised} & \text{если } d > T_{\text{extended}} \\ \text{retracted} & \text{если } d < T_{\text{retracted}} \\ \text{neutral} & \text{иначе} \end{cases}$$

где $T_{\text{extended}} = 0.20$ и $T_{\text{retracted}} = 0.14$ - пороговые значения.

Retract-втянуть

def map_hand_to_screen(self, hand_x, hand_y, frame_width, frame_height):

Математические преобразования:

1. **Нормализация:**

$$x_{\text{norm}} = \frac{x_{\text{hand}}}{W_{\text{frame}}}, y_{\text{norm}} = \frac{y_{\text{hand}}}{H_{\text{frame}}}$$

2. **Зеркальное отображение:**

$$x_{\text{mirrored}} = 1 - x_{\text{norm}}$$

3. **Ограничение зоной отслеживания:**

$$x_{\text{zone}} = \max\left(x_{\min}, \min\left(x_{\max}, x_{\text{mirrored}}\right)\right)$$

4. **Линейное преобразование к экрану:**

$$x_{\text{screen}} = \frac{x_{\text{zone}} - x_{\min}}{x_{\max} - x_{\min}} \cdot W_{\text{screen}}$$

5. **Экспоненциальное сглаживание:**

$$S_t = \alpha \cdot \bar{x} + (1 - \alpha) \cdot S_{t-1}$$

где $\alpha = 0.7$ - коэффициент сглаживания.