

Обработка точечных особенностей

Трекинг, алгоритмы слежения.

Точечная особенность:

Изображения - точечная особенность изображения m - это такая точка изображения, окрестность которой $o(m)$ можно отличить от окрестности любой другой точки изображения $o(n)$ в некоторой другой окрестности особой точки. $o_2(m)$.

Сцены - такая точка сцены M , лежащая на плоском участке поверхности сцены PlaneSegment , изображение окрестности $I(\text{PlaneSegment})$ которой можно отличить от изображений окрестностей всех других точек сцены N из некоторой другой окрестности этой точки $O(M)$.

Точечной особенности сцены должна соответствовать точечная особенность изображений. Обратное неверно: существуют такие особые точки изображения, которым не соответствует никакие особые точки сцены. Такие точки называется ложными особенностями сцены (false feature point).

Задача слежения

Дана последовательность изображений ImageSequence одной и той же сцены S , полученная с движущейся или неподвижной камеры, и набор точечных особенностей $\{N\}$, выделенных в первом кадре последовательности. Для каждой точечной особенности n из $\{N\}$ найти такие точки $n(t)$ на всех изображениях, что их окрестности будут максимально близки к окрестности $n(0)$, с учетом предполагаемой природы искажения ее окрестности и движения точки.

Основные допущения:

- “Малые” изменения от кадра к кадру
- Освещение от кадра к кадру - не меняется
- Особенность - плоская
- Изменения сцены описываются аффинными преобразованиями

Схема А

Этап 1 (Детектирование):

Определить в первом кадре особенности изображения

Этап 2 (Слежение):

Для каждого последующего кадра:

Для каждой особенности $\text{Feature}(i)$:

Найти новое положение особенности в кадре t

Схема В

Этап 1 (Детектирование и оценка)

1. Найти набор особенностей {Features}
2. Определить качество всех особенностей - $Quality(\{Features\})$
3. Оставить только особенности, чье качество выше некоторого заранее или динамически определенного порога, получив множество {GoodFeatures}

Этап 2 (Слежение и оценка)

Для каждого последующего кадра:

1. Найти в текущем кадре новое положение всех особенностей из {GoodFeature} - слежение (tracking)
2. Определить текущее качество всех {GoodFeatures}
3. Оставить только те особенности, чье качество удовлетворяет некоторому критерию
4. Если число отслеживаемых точек падает ниже требуемого, применить детектор к текущему изображению и добавить в {GoodFeatures} новые точки.

Нахождение набора особенностей

Рассмотрим матрицу

$$\mathbf{M} = \begin{bmatrix} \left(\frac{\partial I}{\partial x}\right)^2 & \left(\frac{\partial I}{\partial x}\right)\left(\frac{\partial I}{\partial y}\right) \\ \left(\frac{\partial I}{\partial x}\right)\left(\frac{\partial I}{\partial y}\right) & \left(\frac{\partial I}{\partial y}\right)^2 \end{bmatrix}$$

Если оба ее собственных значения велики, то даже небольшое смещение точки (x,y) в сторону вызывает значительные изменения в яркости. Что и соответствует особенности изображения. Функция отклика угла записывается в следующем виде:

$$R = \det \mathbf{M} - k(\text{trace} \mathbf{M})^2$$

Для снижения влияния шумов на найденные особенности используется сглаживание по Гауссу, но не в самом изображении, а в

картах частных производных: $\left(\frac{\partial I}{\partial x}\right)^2, \left(\frac{\partial I}{\partial y}\right)^2, \left(\frac{\partial I}{\partial x}\right)\left(\frac{\partial I}{\partial y}\right)$

Во многих случаях находится чересчур большое количество углов, из-за чего в дальнейшем их будет сложно отслеживать.

Поэтому вводится ограничение на минимальное расстояние между найденными особенностями, и все лишние отбрасываются.

Математическая формулировка задачи слежения

Пусть $I(x, t)$ - яркость изображения-кадра со временем t в точке x , где x - вектор. Движение изображения (image motion), вдали от границ видимости (occluding boundaries), описывается с помощью уравнения вида: $I(x, t) = I(\delta(x), t+t_1)$ (*), где $\delta(x)$ - движение точки x при переходе от кадра (t) к $(t+t_1)$.

Перемещение особенности от кадра к кадру описывается этим уравнением для всех точек x из окрестности особенности W .

Отметим, что в этом случае полагается, что освещение точки сцены, соответствующей особенности, остается постоянным.

При малых изменениях изображения от кадра к кадру можно считать, что окно особенности просто смещается, и движение $\delta(x)$ принимает вид $\delta(x) = x + d$. Однако при увеличении длительности слежения, изображение точки сцены искажается. Это искажение может быть приближенно описано аффинной трансформацией, поэтому движение точек описывается аффинным преобразованием $\delta(x) = Ax + d$, где A - матрица размерности 2×2 .

Задача трекера заключается в отыскании значения движения $\delta(x)$ для всех точек окна особенности W . Т.к. в реальных условиях (*) никогда строго не выполняется, то ищется такое движение, при котором минимизируется разница между окнами при текущем и будущем положении особенности, т.е. такое $\delta(x)$, при котором достигается минимум

$$e = |I(\delta(x), t+t_1) - I(x, t)|,$$

или

$$e = \min \sum_W (I(\delta(x), t + t_1) - I(x, t))^2,$$

если норма разности изображений L_2

Развитие алгоритмов слежения

Все современные алгоритмы слежения за особенностями опираются на работу 1981 году Лукаса и Канаде. В 1991 году математическая формулировка этого алгоритма была изменена, и стала основой для всех последующих обобщений с учетом аффинных искажений окрестности и освещенности. Путем замены соответствующих переменных на константы любой из них превращается в обычный алгоритм Lucas-Kanade.

- ***Lucas-Kanade*** - особенность считается только смещающейся, без искажений
- ***Tomasi-Kanade*** - переформулирование Lucas-Kanade. Движение считается смещением, и рассчитывается путем итеративного решения построенной системы линейных уравнений.
- ***Shi-Tomasi-Kanade*** - учитывает аффинные искажения особенности
- ***Jin-Favaro-Soatto*** - модификация Shi-Tomasi-Kanade с учетом аффинных изменений освещенности особенности

Пример трекинга особенностей

