

图像处理作业 3 —— 视频去抖动

本次作业中使用特征提取、特征追踪和帧间运动的平滑实现视频去抖动.

1 GFTT 特征点提取

1.1 基于特征值的角点检测

设 f 是一副图像, w 是权函数, $f(S, T)$ 是一小块图像, $f(S + x, T + y)$ 是尺寸相同但移动了 (x, y) 的小块图像, 两幅图像差的平方加权和

$$C(x, y) = \sum_{s \in S} \sum_{t \in T} w(s, t) [f(s + x, t + y) - f(s, t)]^2$$

对表达式进行 Taylor 展开, 记 $f_x = \partial f / \partial x$, $f_y = \partial f / \partial t$, 则

$$C(x, y) = \sum_{s \in S} \sum_{t \in T} w(s, t) [xf_x(s, t) + yf_y(s, t) + \varepsilon]^2 = [x, y] \mathbf{M} [x, y]^T$$

其中

$$\mathbf{M} = \sum_{s \in S} \sum_{t \in T} w(s, t) \mathbf{A}(s, t), \quad \mathbf{A} = \begin{bmatrix} f_x^2 & f_x f_y \\ f_x f_y & f_y^2 \end{bmatrix}$$

\mathbf{M} 为 Harris 矩阵, 这个矩阵的两个特征值 λ_1, λ_2 反映了区域特征. 当两个特征值均较大时, 表明区域内包含一个角点或孤点.

1.2 GFTT 筛选特征点

直接使用 Harris 矩阵特征值提取角点时, 若阈值设置不当容易使得特征点过多或过密, 对特征追踪的效率和效果均会产生影响, 因此需要对特征点的数量和质量进行控制. GFTT[3] 说明了易于进行追踪的特征点应具有的特征, 并可使用以下算法对通过 Harris 矩阵得到的特征点进行筛选.

- 对于每个特征点 i , 其坐标为 (x_i, y_i) , Harris 矩阵的最小特征值 $\lambda_{i, \min}$
- 在 3×3 范围内进行非极大值抑制, 以缓解部分区域特征点过密的情况
- 对于特征点 i , 若 $\lambda_{i, \min} < C \max_j \lambda_{j, \min}$, 则去除 i , 其中 C 是衡量特征点质量的阈值
- 对于剩余的特征点, 若存在 i, j 使得 $\lambda_{i, \min} > \lambda_{j, \min}$ 且 $\|(x_i, y_i) - (x_j, y_j)\| \leq d$, 则去除 j , 从而进一步提高特征点质量, 降低特征点密度, 其中 d 是可调节的距离阈值
- 按照 λ_{\min} 降序排序所有特征点

直接用 Harris 矩阵得到的特征点和经过 GFTT 筛选的特征点对比见 fig. 1



图 1: 左侧为使用 Harris 矩阵提取的特征点, 右侧为经过 GFTT 所筛选出的结果.

2 LK 光流法特征追踪

在视频处理中, 相邻两帧的变化相差并不大, 因此只在每个检测到的特征周围搜索小邻域 (特征追踪) 比全局性的特征匹配更为合适. 光流法通过估计某个特征点或像素在图像上的移动速度以追踪该点在下一帧图像上的位置. Lucas-Kanade(LK) 光流法 [2] 是求解光流的一种重要方法.

2.1 LK 光流法

在 LK 光流法中假设图像具有以下特征

- (1) 目标像素在帧间运动时像素灰度不发生变化
- (2) 目标像素运动速度较慢, 可以在下一帧的小邻域内检测到
- (3) 目标像素的小邻域内的点具有相似的运动

用 $I(x, y, t)$ 表示连续的视频. 根据 (1), 对于某个特征点 (x, y, t) ,

$$I(x + dx, y + dy, t + dt) = I(x, y, t)$$

根据 (2), 可以对左侧进行 Taylor 展开并略去高阶量

$$I(x, y, t) + \frac{\partial I}{\partial x}dx + \frac{\partial I}{\partial y}dy + \frac{\partial I}{\partial t}dt = I(x, y, t)$$

也即

$$\frac{\partial I}{\partial x}u_x + \frac{\partial I}{\partial y}u_y + \frac{\partial I}{\partial t} = 0$$

其中 u_x 和 u_y 为像素运动速度, 根据 (3), 可以对目标像素的小邻域均列出上述方程, 并利用最小二乘或 RANSAC 等方法求得 u_x 和 u_y .

2.2 LK 光流法的改进

由于假设 LK 法的 (2) 并不能常常满足, 需要能够处理较大尺度运动的方法. Pyramid LK 法 [1] 通过多尺度降采样图像上运行 LK 的方法 (fig. 2), 实现对更快更长的运动的跟踪

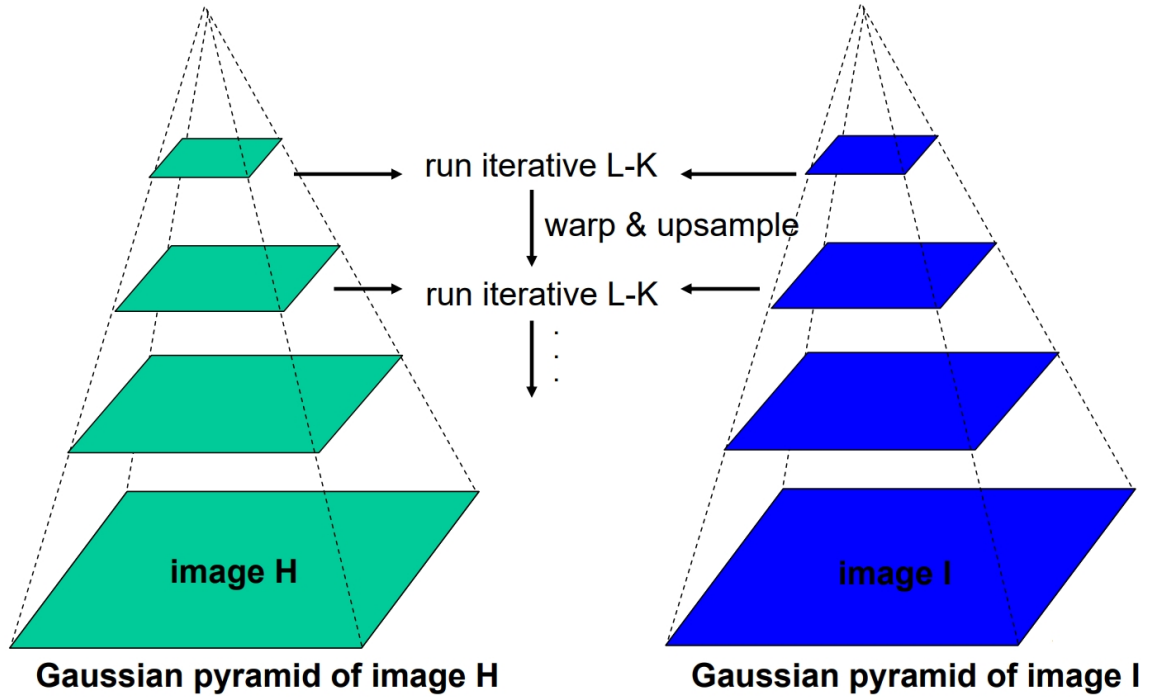


图 2: Pyramid LK 光流法图示, 来自CSE576

3 运动平滑化

设视频的图像序列为 I_0, I_1, I_2, \dots 令 $T_{i,j} = [\delta x, \delta y, \delta \alpha]_{i,j}$ 表示从 I_i 到 I_j 的平移和旋转, 也即 I_i 中坐标为 (x, y) 的点在 I_j 中的坐标为

$$\begin{bmatrix} \cos \delta \alpha & \sin \delta \alpha & \delta x \\ -\sin \delta \alpha & \cos \delta \alpha & \delta y \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

对任意的 i , $T_{i,i+1}$ 可以由 LK 光流法得到的相邻两帧的匹配特征点对求出, 对于 $j > i + 1$, 取

$$T_{i,j} = \sum_{k=i}^{j-1} T_{k,k+1}$$

令 $S_i = T_{0,i}$ 表示从 I_0 到 I_i 的累积变换, 只需对 S_i 进行平滑化, 即可去除视频的抖动. 取一个模糊核 w 与 S 进行卷积, 得到平滑后的 $\tilde{S} = w \star S$. 只需将 $T_{i-1,i} + (\tilde{S}_i - S_i)$ 作用在 I_i 上, 即可实现平滑化.

fig. 3展现了对于某个视频所计算出的 T, S, \tilde{S} 和最终作用在各帧上的变换.

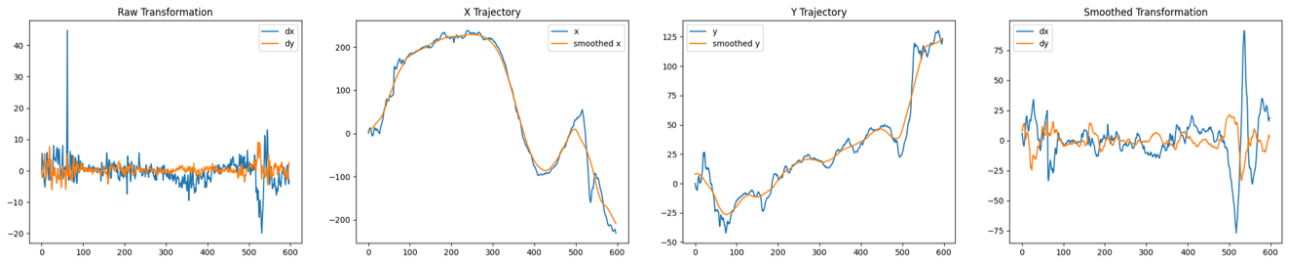


图 3: 对视频文件 `demo/demo.mp4` 进行运动平滑化的中间结果. 自左至右分别为 (a) $T_{i,i+1}$ 的 $\delta x, \delta y$ 项; (b) S 和 \tilde{S} 的 δx 项; (c) S 和 \tilde{S} 的 δy 项; (d) 最终作用在每一帧上的变换的 $\delta x, \delta y$ 项. 去抖动后的视频文件为 `demo/output.avi`

参考文献

- [1] Jean-Yves Bouguet et al. Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm. *Intel corporation*, 5(1-10):4, 2001.
- [2] Bruce D Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. In *IJCAI'81: 7th international joint conference on Artificial intelligence*, volume 2, pages 674–679, 1981.
- [3] Jianbo Shi et al. Good features to track. In *1994 Proceedings of IEEE conference on computer vision and pattern recognition*, pages 593–600. IEEE, 1994.