Custom kernel ⟶ edges ⟶ $\Sigma = 0$

Canny Algo ⟶ باقي ⟶ zero crossing

Corner ⟶ 2d changes (all regions)

- Find image feature ⟶ Feature detection
- describe region around feature ⟶ Feature description
  ↳ neighbours

(1) Harris Corner detection

① window, large variation in intensity (x,y)

② score ③ threeshold

① $E(u,v) = \sum w(x,y) \left[I(x+u, y+v) - I(x,y) \right]^2$

② $R = \underbrace{\det(m)}_{\lambda_1 \lambda_2} - k \underbrace{\text{trace}(m)^2}_{\lambda_1+\lambda_2}$

λ_1, λ_2 eigen values of m

③ threeshold

$R \longrightarrow$ small $\longrightarrow$ flat

$R < 0 \longrightarrow$ edge

$\Rightarrow R \longrightarrow$ large $\longrightarrow$ Corner

`cv2.cornerHarris(img, blockSize, ksize, k)`

- img: gray float32
- blockSize: neighbourhood
- ksize: Sobel derivative
- k: harris detector

---

(2) Shi-Tomasi Corner detection

`cv2.goodFeatureToTrack(img, N, x, y)`

- img ⟶ grayscale
- N ⟶ max nom. of Corners
- X ⟶ Specify nom of Corners (0–1)
- y ⟶ minimum euclidean distance between Corners