

Good features to track(Shi-Tomasi)

Seleção de pontos de interesse robustos para rastreamento visual

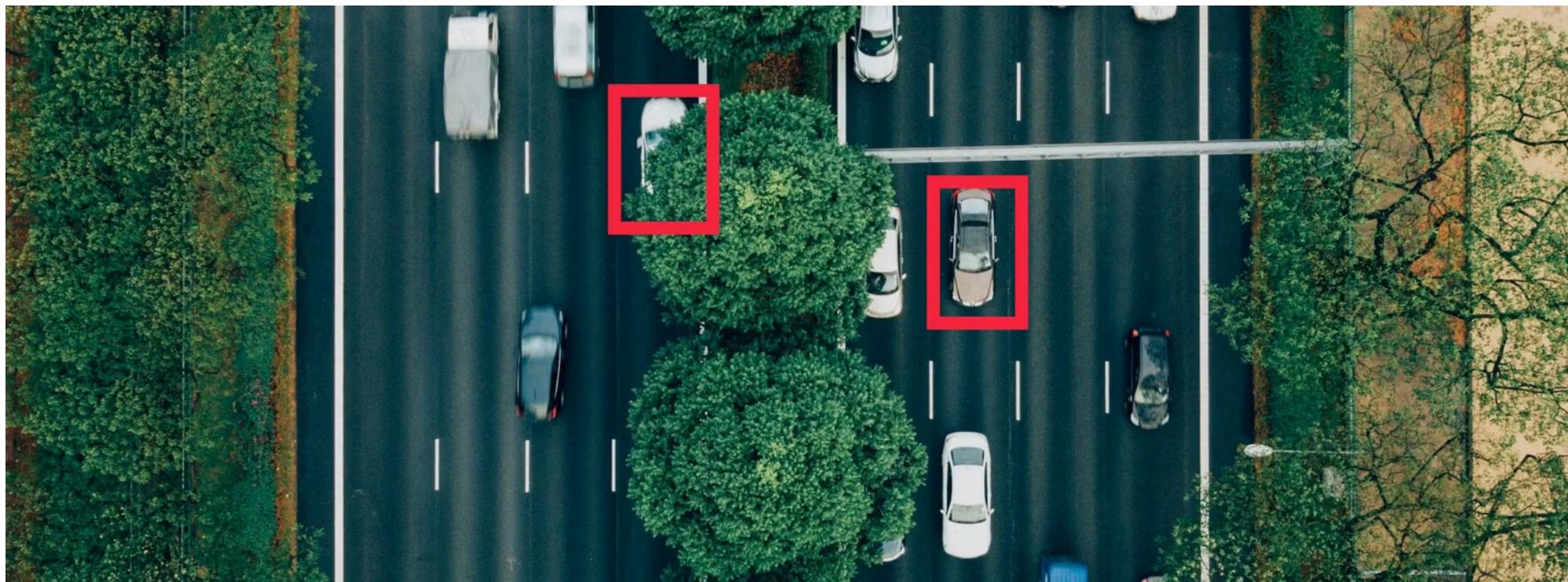
Diogo de Souza Alves

[diogo_souza@discente.ufg
.br](mailto:diogo_souza@discente.ufg.br)



Good features to track

- O problema:
 - Dadas imagens sequenciais, rastrear confiavelmente as mesmas features (pontos de interesse) ao longo da sequência, avaliando constantemente a qualidade de cada ponto.
 - Se uma feature ficar ruim (oculta, deformada, sumir), o sistema descarta.
 - Medimos o quanto rastreável é um ponto e mantemos apenas aqueles que são bons e que se mantêm confiáveis





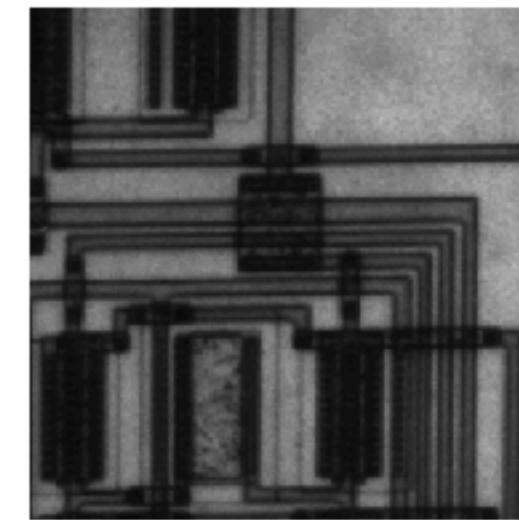
Como usamos matematicamente:

$$M = \sum_{w(x,y)} \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}$$

- **Structure tensor(M)**

Constrói essa matriz 2x2 que descreve como os gradientes (I_x , I_y) se parecem na vizinhança daquele pixel.

Original Image



Gradient Magnitude



- **Detector Harris**

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

$$\det(M) = \lambda_1 \lambda_2$$

$$\text{trace}(M) = \lambda_1 + \lambda_2$$

Projetada para que o valor de R nos diga automaticamente que tipo de região estamos olhando, em 3 casos:

- **Caso 1(Região Plana)**

- Ambos lambdas são pequenos.
- $\det(M) \sim 0$ e $\text{trace}(M) \sim 0$.
- Resultado: R é pequeno (próximo de 0).

- **Caso 2(Borda)**

- Um valor de lambda é alto e o outro é pequeno.
- $\det(M) \sim 0$ (pequeno) e $\text{trace}(M)$ será grande.
- Resultado: R é um número negativo grande.

- **Caso 3(Canto)**

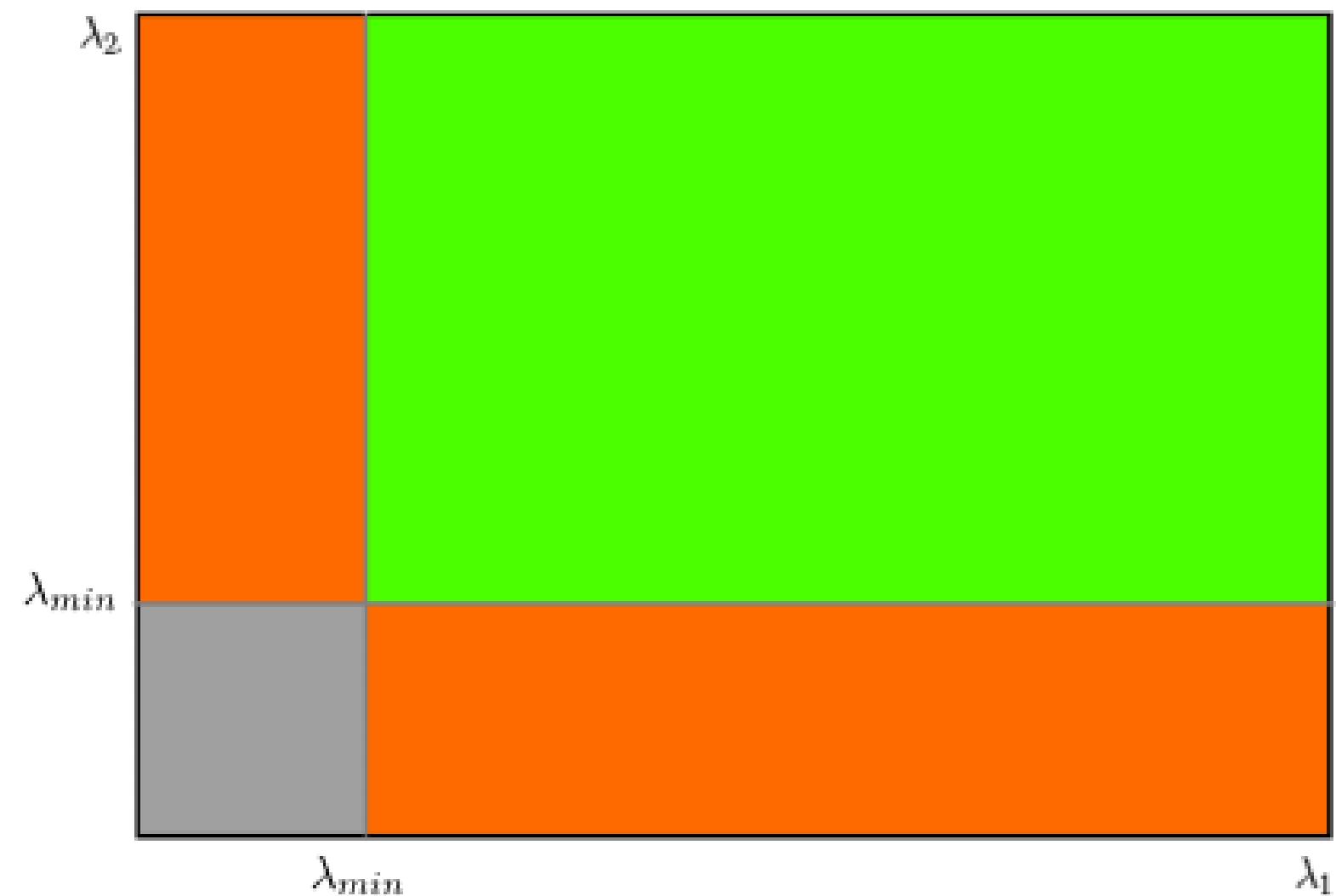
- Ambos lambdas são grandes.
- $\det(M)$ e $\text{trace}(M)$ são números positivos grandes.
- Resultado: R é um número positivo grande.



- **Detector Shi-Tomasi(GFTT)**

$$R = \min(\lambda_1, \lambda_2)$$

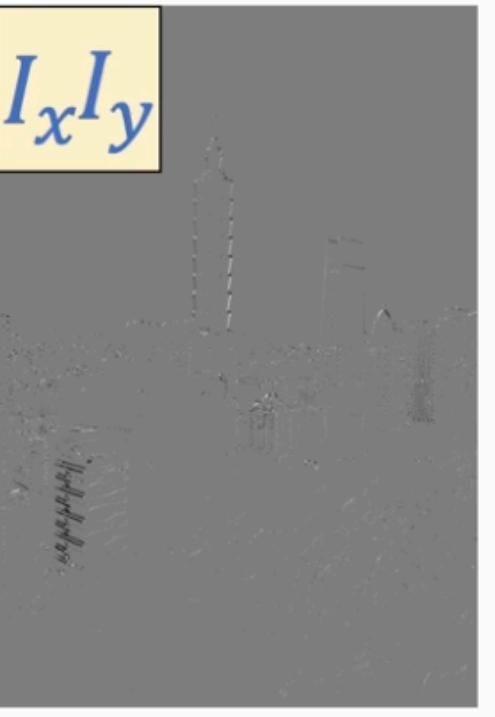
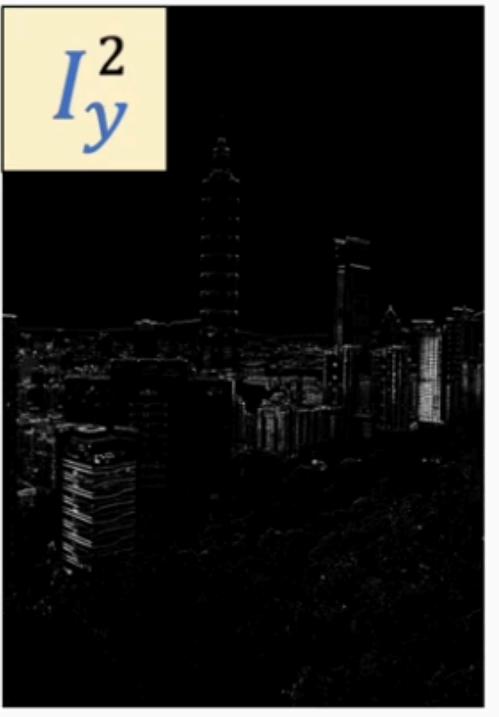
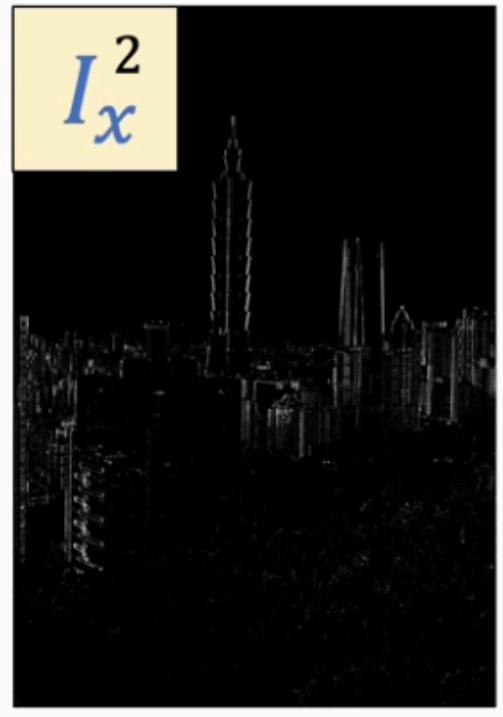
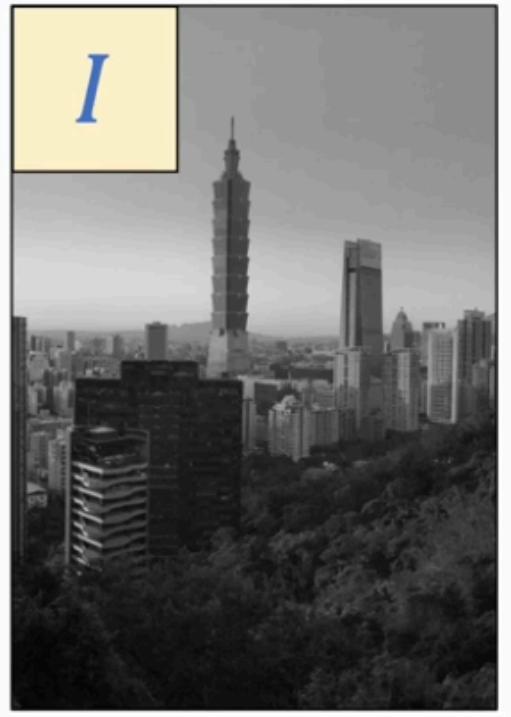
- Calcula os autovalores de ambos os lambdas da matriz M, e a pontuação R é simplesmente o menor dos dois
- Descarta Bordas por serem ambíguas para rastreamento
- Limiar de Qualidade: Descarta todos os cantos abaixo de um limiar mínimo .
- Supressão Não Máxima: Garante que os pontos selecionados estejam a uma distância mínima uns dos outros, evitando aglomerados de "features" no mesmo canto.



- A Região Verde (Cantos - Aceitos)
- A Região Laranja (Bordas - Rejeitadas)
- A Região Cinza (Plana - Rejeitada)

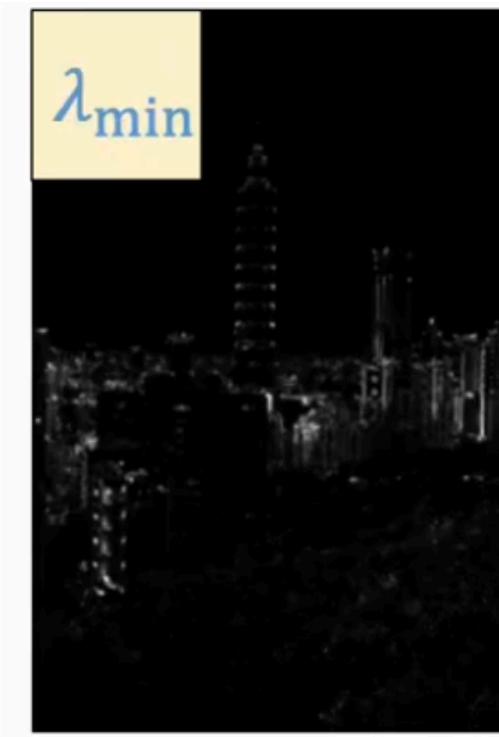
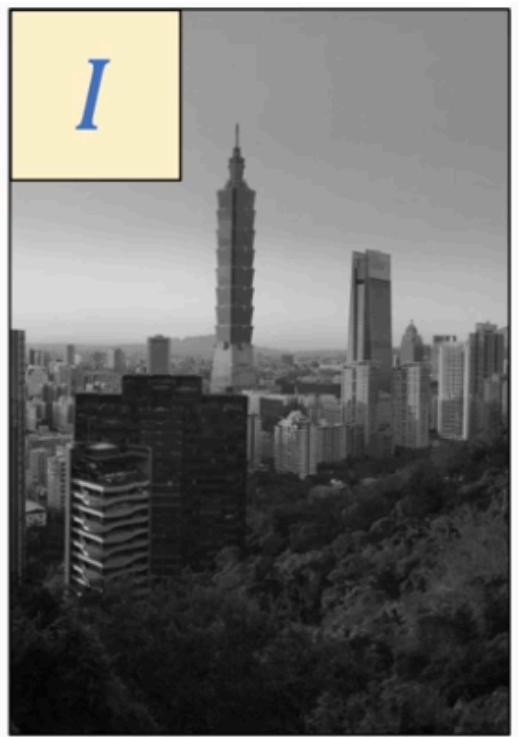
- Detector Shi-Tomasi(GFTT)

$$R = \min(\lambda_1, \lambda_2)$$



$$E(u, v) = \text{auto-correlation function}$$

$$[u \ v] \begin{bmatrix} \sum_{(x,y) \in W} I_x^2 & \sum_{(x,y) \in W} I_x I_y \\ \sum_{(x,y) \in W} I_x I_y & \sum_{(x,y) \in W} I_y^2 \end{bmatrix} [u \ v]$$

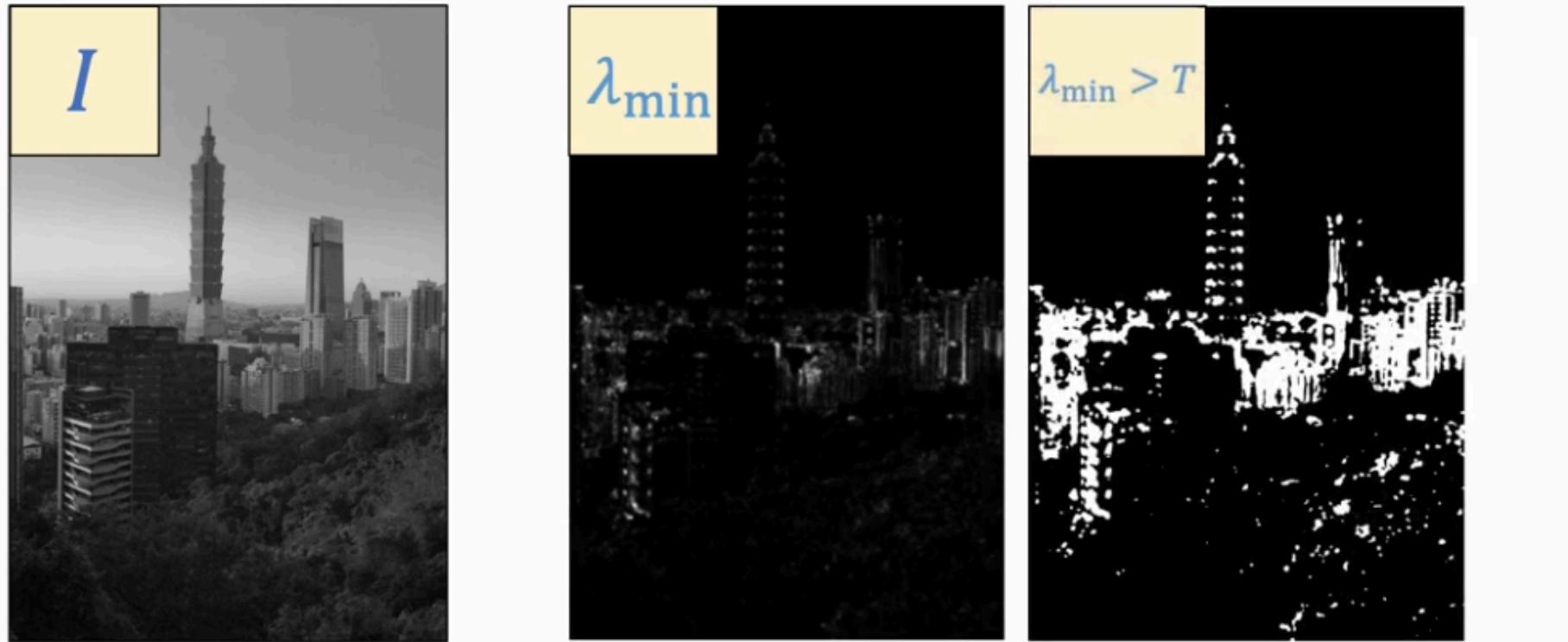


$$E(u, v) = \text{auto-correlation function}$$

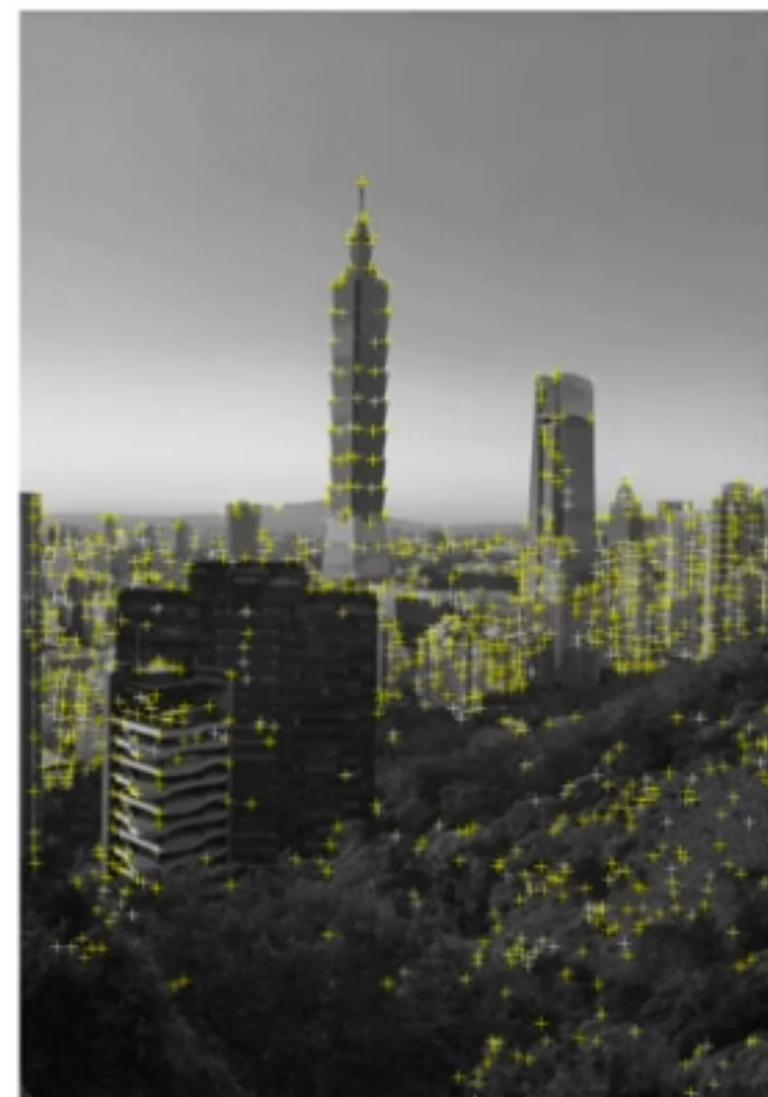
$$[u \ v] \begin{bmatrix} G_\sigma * I_x^2 & G_\sigma * I_x I_y \\ G_\sigma * I_x I_y & G_\sigma * I_y^2 \end{bmatrix} [u \ v]$$



- Detector Shi-Tomasi(GFTT)
 $R = \min(\lambda_1, \lambda_2)$

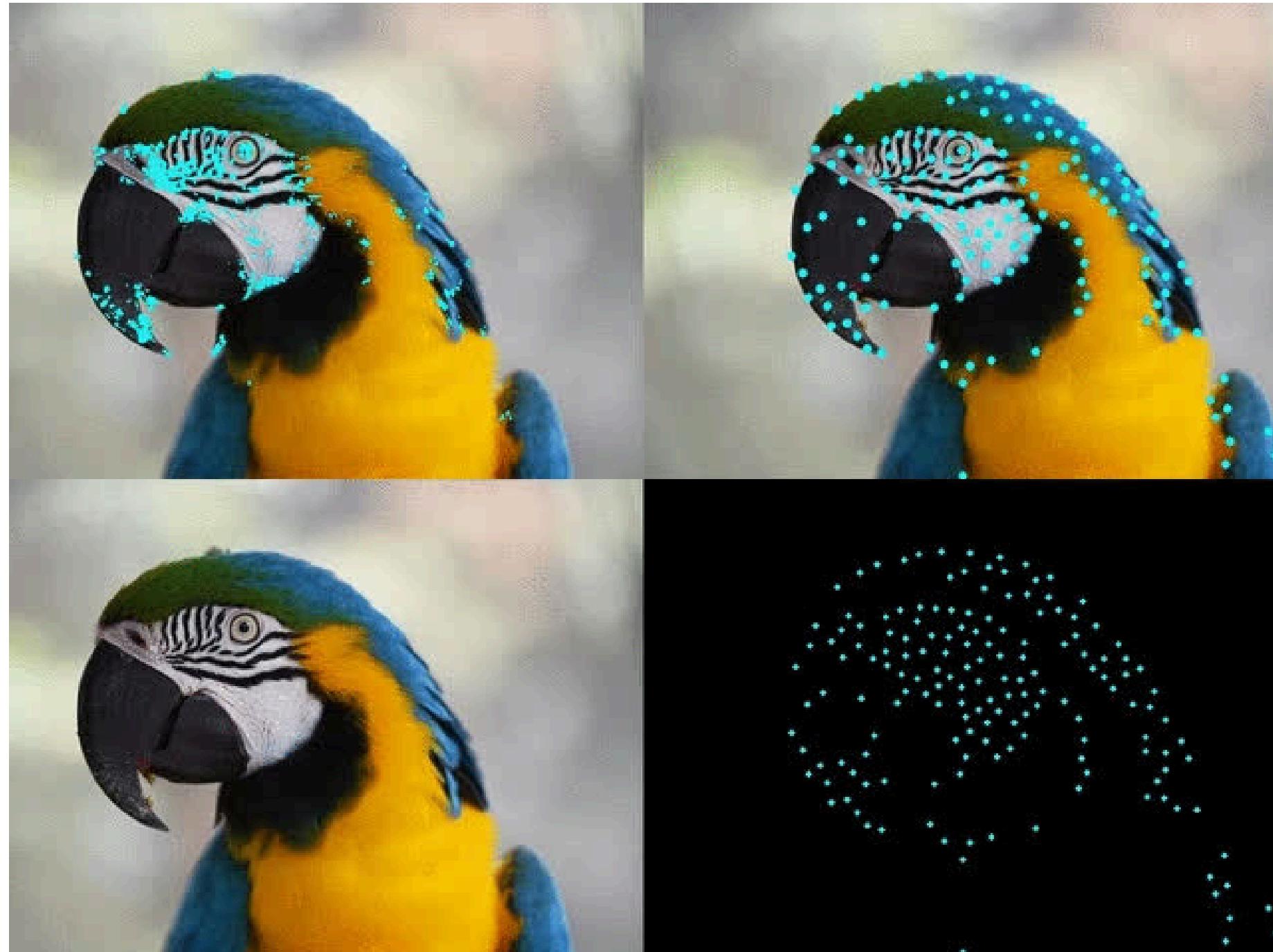


Aplicamos Non-Maximum Suppression (NMS)





Aplicando os algoritmos



Canto superior esquerdo: Harris, Canto superior direito: Shi-Tomasi, Canto inferior esquerdo: Original



Shi-tomasi e Lucas-Kanade

- **shi-tomasi**

- Dectecção
- Analise um único frame e encontra os cantos mais fortes e estáveis
- Uma lista de coodernadas(x,y) de “bons pontos”

- **Kanade-Lucas-Tomasi(KLT)**

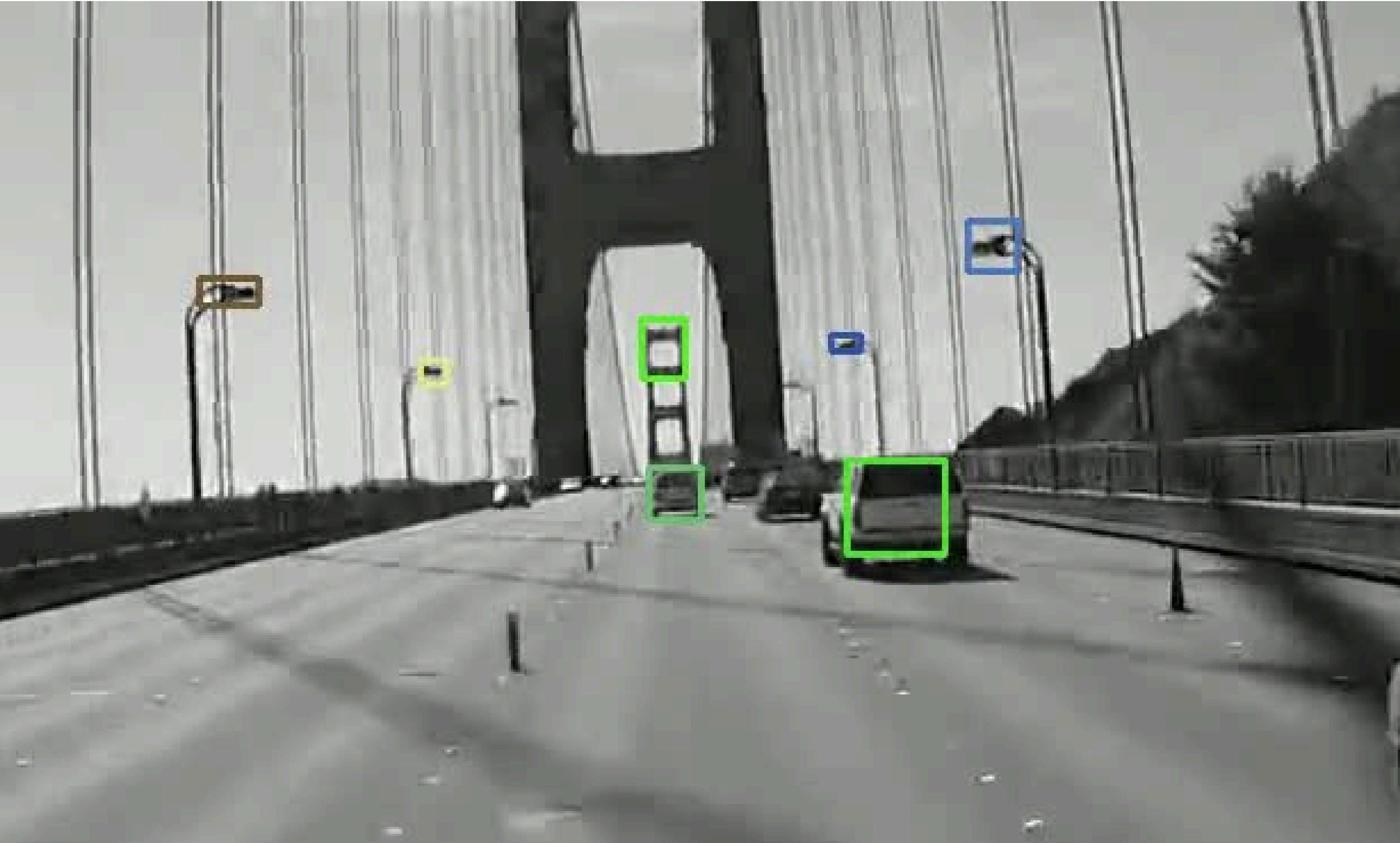
- Rastreamento
- Compara dois frames para calcular o movimento dos pontos
- Um vetor de movimento(u,v) para cada ponto de entrada

- **Resolve o problema**

Ao definir a pontuação como o menor dos dois autovalores, Tomasi garantiu que qualquer ponto selecionado tivesse ambos os autovalores grandes.

- O GFTT atua como o filtro de entrada perfeito para o KLT.
- Ele garante que o KLT só receba pontos onde a matriz M é robusta e matematicamente estável para inversão.
- Em outras palavras: O GFTT encontra os pontos que o KLT consegue rastrear.

KLT Tracking Algorithm

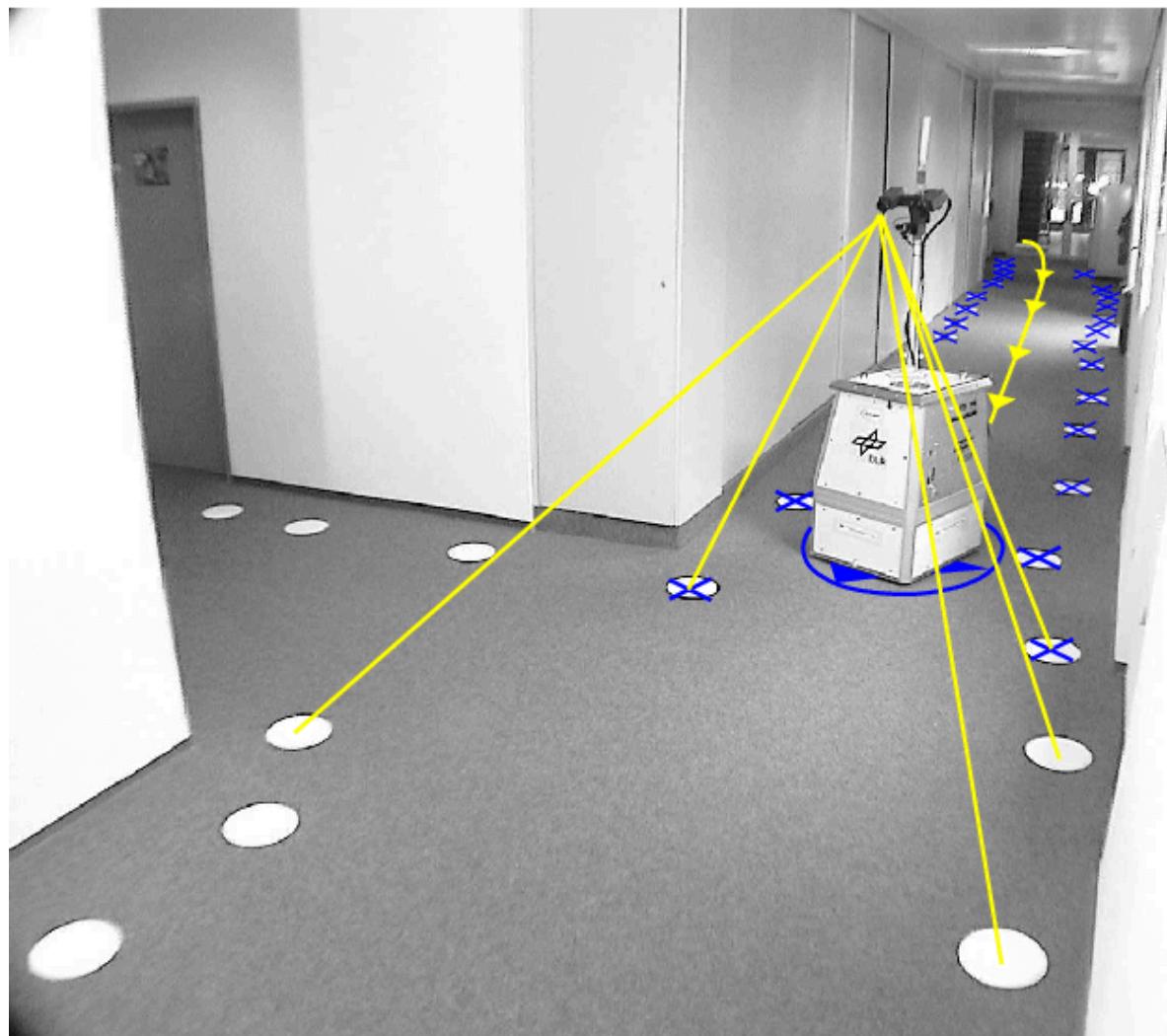


Aplicações na Robótica



SLAM (simultaneous localization and mapping)

- O que é: É o processo de construir um mapa 3D de um ambiente desconhecido e, ao mesmo tempo, saber onde você está dentro desse mapa.
- Relação com Shi-Tomasi: O SLAM usa os pontos do Shi-Tomasi como os "tijolos" do mapa. Os cantos encontrados pelo GFTT são rastreados (pela VO) e, em seguida, triangulados no espaço para se tornarem os "marcos" (pontos) do mapa 3D.



O objetivo do SLAM é criar um mapa consistente do ambiente enquanto o robô navega por ele, mesmo que não haja um mapa pré-existente



Estabilização de câmeras e drones

- **O que é:** O processo de remover digitalmente a "tremedeira" (vibração) de um vídeo para que ele pareça suave.
- **Relação com Shi-Tomasi:** O software identifica pontos GFTT no fundo (cenário que deveria estar parado). Se esses pontos "vibram" na imagem, o algoritmo sabe que foi a câmera que tremeu. Ele então move o frame na direção oposta para cancelar a vibração antes de gravá-lo.

Navegação autônoma

- **O que é:** O processo de remover digitalmente a "tremedeira" (vibração) de um vídeo para que ele pareça suave.
- **Relação com Shi-Tomasi:** O software identifica pontos GFTT no fundo (cenário que deveria estar parado). Se esses pontos "vibram" na imagem, o algoritmo sabe que foi a câmera que tremeu. Ele então move o frame na direção oposta para cancelar a vibração antes de gravá-lo.

Obrigado!

diogo_souza@discente.ufg.br

Dúvidas ou sugestões?



INSTITUTO DE
INFORMÁTICA
UFG

