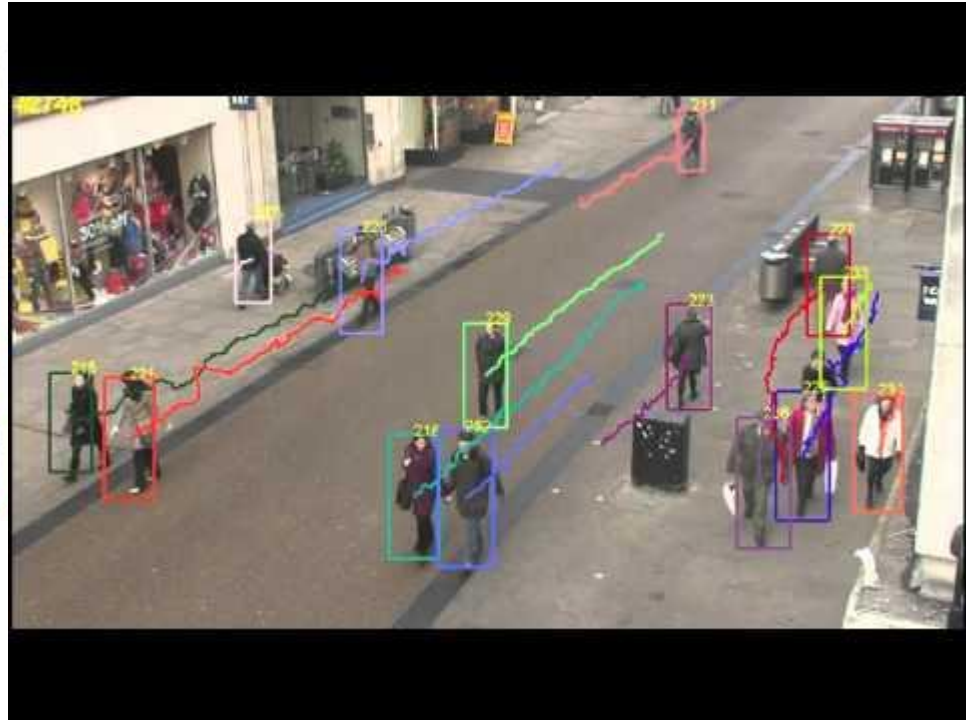
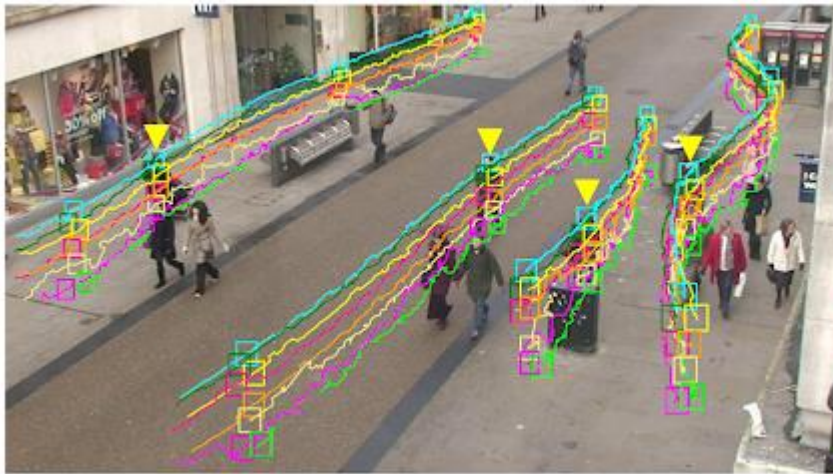


Rastreio/tracking

Rastrear é evidenciar o rastro ou a pista de um ou mais objetos em um vídeo, determinando a/as trajetória/as ao longo do tempo.

O rastreio de objetos em imagens é uma operação muito aplicada em sistemas de monitoramento (surveillance)



Rastreio (*simplório*)

A proposta é utilizar os conhecimentos sobre Template Matching e biblioteca OpenCv discutidos na última aula (links na sala-BBB) aplicando métodos.

cvLoadImage, cvCreateImage, cvMatchTemplate
cvMinMaxLoc, cvRectangle, cvNamedWindow
cvShowImage, etc...

Deseja-se a aplicação do método melhor método de casamento de template como ferramenta para uma estratégia funcional (ainda que simplória) de rastreamento de objeto na cena.

Veja o seguinte tutorial do OpenCV:

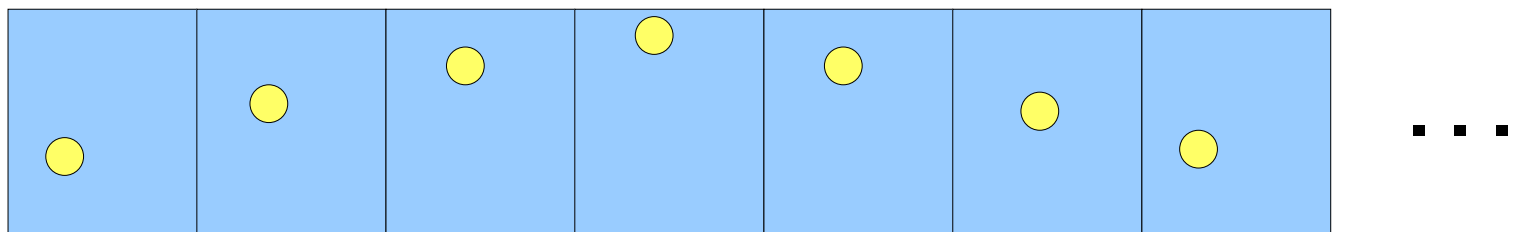
https://docs.opencv.org/4.5.2/d4/dc6/tutorial_py_template_matching.html

A ideia é realizar uma estratégia (simplória) que aplica Template Matching para simular um rastreamento.

Passos da Tarefa:

A) Faça um vídeo "caseiro" de 10 segundos (300 quadros) com sua webcam ou celular, exibindo o deslocamento lento de um objeto (bem enquadrado na cena) sobre um cenário de fundo constante e câmera fixada/estática, sem autofocus (função de foco automático desligada).

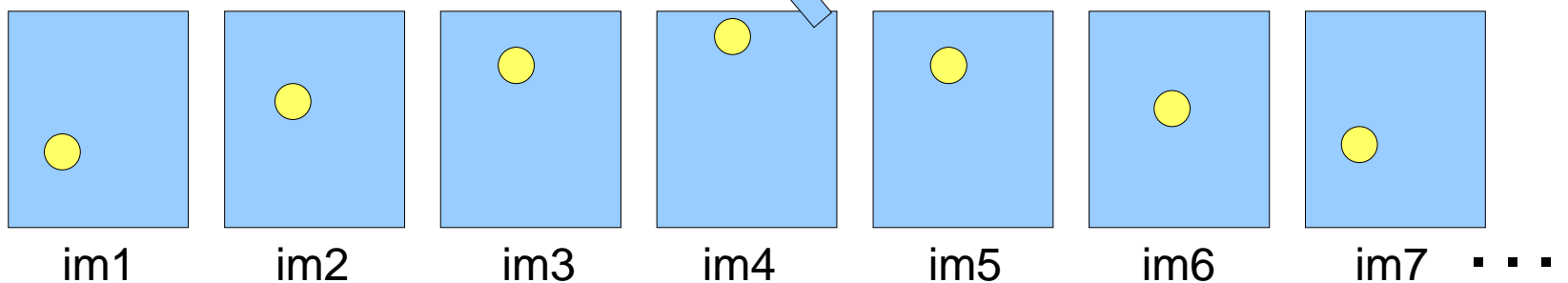
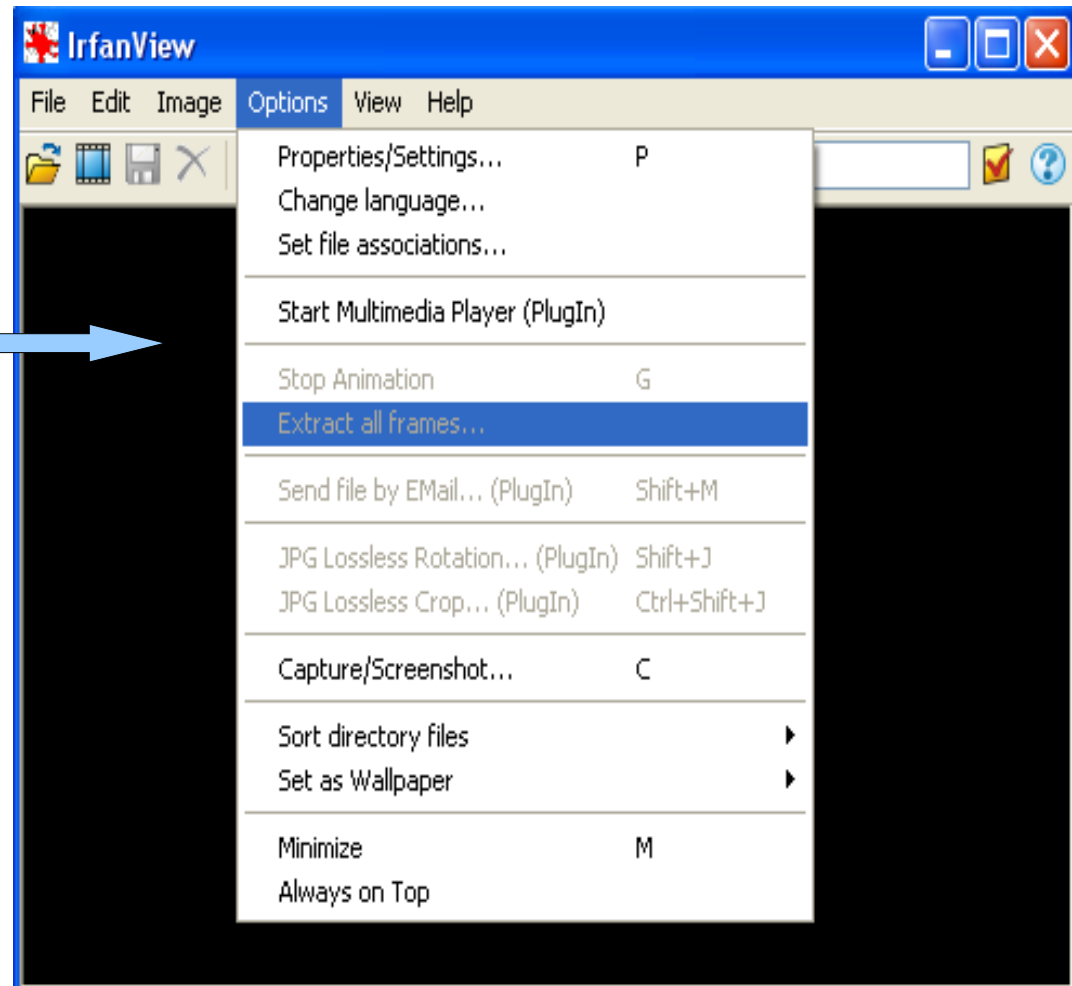
Utilize uma resolução mínima de 640X480 (VGA);



B) Separe os quadros utilizando, por exemplo, o software gratuito Irfanview;

C) Salve esses quadros como imagens em tons de cinza.

Esses quadros corresponderão ao conjunto de imagens: im1, im2, im3,...



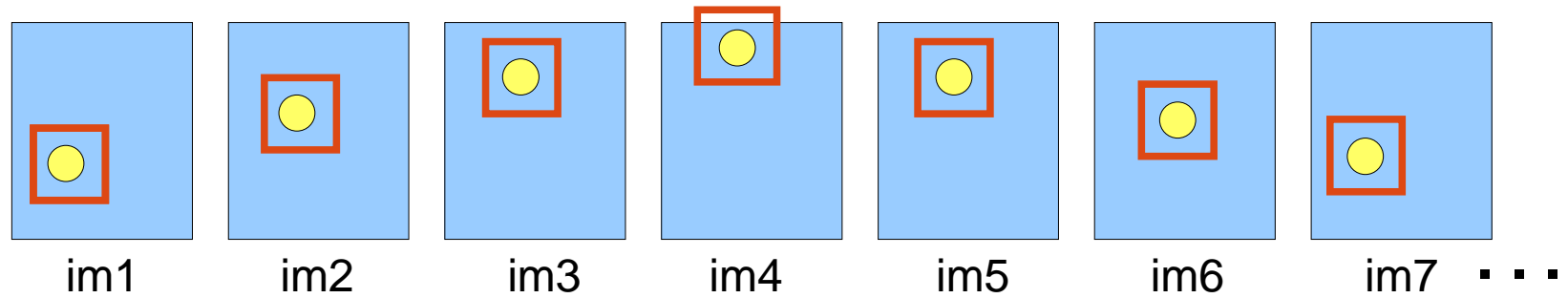
D) Utilize uma amostra da primeira imagem `im1` para obter um recorte contendo o objeto a ser rastreado, o qual será o *template* a ser rastreado a cada imagem.

E) Realize testes e identifique o melhor método de casamento a ser utilizado como parâmetro para `cvMatchTemplate`. Os métodos são:

- `'cv.TM_CCOEFF'`,
- `'cv.TM_CCOEFF_NORMED'`,
- `'cv.TM_CCORR'`,
- `'cv.TM_CCORR_NORMED'`,
- `'cv.TM_SQDIFF'` e
- `'cv.TM_SQDIFF_NORMED'`

Faça isso com imagens "caseiras" geradas por você com sua webcam ou celular. Essas imagens têm que ser capturadas no mesmo ambiente do vídeo solicitado no item A.

F) Construa um vídeo de saída com as imagens identificando o objeto rastreado.



G) Construa um vídeo de saída com as imagens de respostas do `cvTemplateMatching` utilizado nas detecções de casamento do template (objeto rastreado).



H) Construa uma tabela indicando cada imagem e respectivo valor máximo e mínimo detectado para o casamento;

I) Execute os mesmos passos para uma sequência de quadros com uma cena mais complexa. Exemplo: o seu mascote se movendo em um cenário de fundo com mais informações visuais.

J) Entregue um relatório contendo:

- A solução da tarefa em todas as suas etapas.
- Identificação da equipe (Máximo 3 membros).
- Scripts (terminação ".py") com comentários explicativos nos trechos importantes

Entrega/entrevista: veja no moodle, equipes de no máximo dois membros

Implementações em Python-OpenCv

Entregar:

Relatório contendo algum embasamento teórico, descrevendo a metodologia do experimento, resultados, análise, conclusão e bibliografia. Insira ilustrações no relatório, analise e discuta os resultados.

Dados mínimos a serem fornecidos/exibidos: vídeo de entrada, vídeo de saída (rastreo marcado nos frames), vídeo com as respectivas imagens de resposta ao cvMatchTemplate; Tabela indicando cada imagem e respectivo valor máximo e mínimo detectado para o casamento; anexe os códigos-fontes utilizados.

Referências:

Tutoriais sobre OpenCV, Python, bibliografia da disciplina

Sites:

docs.opencv.org/4.5.2/d4/dc6/tutorial_py_template_matching.html

docs.opencv.org/3.4/de/da9/tutorial_template_matching.html