

## Trabalho Prático

### Resumo

Com este trabalho pretende-se dar aos alunos a oportunidade de aplicar alguns dos conceitos de processamento e análise de imagem abordados nas aulas. Os alunos deverão desenvolver um programa em C ou C++, que opere sobre vídeo disponibilizado para este trabalho prático, de modo a obter o conjunto de informação desejada (ver abaixo). Este trabalho culminará na entrega do código fonte desenvolvido, para além da obrigatória defesa oral do trabalho realizado.

### Realização do trabalho prático

No ficheiro “VC-TP.zip” é apresentada a respetiva descrição do trabalho e objetivos, contendo ainda:

- Vídeos “video1.mp4” e “video2.mp4” a analisar;
- Código fonte com exemplo de leitura/captura e exibição de vídeo com recurso à biblioteca OpenCV.

O trabalho deverá ser desenvolvido em grupo, sendo que os **grupos deverão ser constituídos por 4 elementos**.

Espera-se que os alunos utilizem os conhecimentos adquiridos ao longo das aulas de Visão por Computador, e desenvolvam as funcionalidades necessárias ao cumprimento dos objetivos.

Note que todo o código deverá ser realizado em linguagem C ou C++, podendo o aluno utilizar (para além das funções OpenCV identificadas no ficheiro “CodigoExemplo.cpp”, ou funções similares, dependendo da versão do OpenCV) **até mais 3 funções ou instâncias de classes da biblioteca OpenCV**. Por exemplo, a seguinte linha de código atingiria este limite, por usar `cv::rectangle()`, `cv::Point()` e `cv::Scalar()`:

```
cv::rectangle(frame, cv::Point(10, 2), cv::Point(100,20), cv::Scalar(255,255,255), -1);
```

Não é permitida a utilização de: bibliotecas externas de processamento de imagem não abordadas nas aulas; código disponibilizado em repositórios (como por exemplo, GitHub); bem como qualquer outro código cuja autoria não seja dos elementos que constituem o grupo.

O trabalho deverá ainda ser apresentado/defendido por todos os elementos do grupo numa **apresentação/defesa oral**, sendo ainda colocadas **questões sobre o trabalho** aos vários elementos do grupo. Não é necessária a realização de qualquer relatório ou apresentação PowerPoint. A apresentação/defesa deverá incidir sobre a estratégia seguida e respetiva implementação (código fonte), as dificuldades sentidas e a resolução encontrada para as mesmas. Espera-se que os alunos dominem todo o código fonte utilizado pelo grupo para a resolução do trabalho prático.

### Objetivo

Este trabalho tem como objetivo o desenvolvimento de um programa em linguagem C, ou C++, que permita a automática quantificação de dinheiro em vídeos. Neste sentido, o programa deverá efetuar a leitura de um dado ficheiro indicado pelo utilizador (em formato \*.mp4), identificar todas as moedas visíveis ao longo do vídeo, e apresentar o respetivo somatório. Mais ainda, o seguinte conjunto de informação deverá ser disponibilizado durante e/ou após o processamento do vídeo (sobre este):

- Número total de moedas observadas ao longo do vídeo;

- Número de moedas por tipo (1, 2, 5, 10, 20 ou 50 cent., e 1 ou 2 euros);
- Área e perímetro (em pixéis) de todas as moedas observadas ao longo do vídeo;
- Desenho, sobre a imagem a exibir em vídeo, da localização (área delimitadora) e centro de gravidade de cada moeda, bem como indicação do seu respetivo tipo.

Tenha ainda em consideração que:

- Os vídeos têm todos uma resolução de 1280x720;
- Os vídeos têm uma *frame rate* de 30 fps.

### Especificação do Sistema

O sistema de visão por computador deverá contemplar:

- Implementação de técnicas de segmentação (por tonalidade e/ou brilho);
- Implementação de técnicas para melhoramento de imagem (e.g., remoção de ruído);
- Implementação de técnicas de análise de imagem que permitam determinar:
  - Área;
  - Caixa delimitadora;
  - Circularidade dos objetos segmentados;
  - Centro de massa (centroide); entre outras informações que o grupo entenda como relevantes.
- Algoritmos que permitam distinguir as diferentes moedas.

### Avaliação

Serão tomados como critérios de avaliação os seguintes fatores:

- Qualidade do programa:
  - desenvolvimento das funcionalidades descritas no enunciado do trabalho;
  - nível de otimização das funções implementadas;
  - funcionamento correto do programa;
  - valor acrescentado<sup>1</sup>.
- Qualidade do código e respetivos comentários, bem como da apresentação oral:
  - descrição correta e completa da estrutura do programa;
  - descrição das técnicas aplicadas no desenvolvimento das funcionalidades.
- Respeito pelas regras de entrega do trabalho.
- Nível de conhecimento que o estudante possui sobre o trabalho desenvolvido.

A natureza coletiva da realização de um trabalho em grupo não prejudica o facto de a avaliação ser individual para cada um dos elementos do grupo.

Cada elemento do grupo deve demonstrar o conhecimento de todo o código submetido.

---

<sup>1</sup> Por valor acrescentado entende-se a forma como o trabalho se destaca (positivamente) dos restantes.

## NOTAS IMPORTANTES:

- A defesa do trabalho é obrigatória. A sua não realização implica a atribuição de 0 (zero) valores à componente prática.
- Para a defesa, no dia agendado, o grupo deve trazer pelo menos um PC com o código editável e compilável. No entanto, recomenda-se que tragam mais que um computador para acautelar eventuais problemas.
- O código fonte para a apresentação/defesa deverá ser o mesmo que submeteram para a avaliação, mas deverão previamente retirar-lhe todos os comentários.

## Prazos

A realização do trabalho pressupõe a entrega dos ficheiros com o código fonte, em formato digital. Não submeta a pasta de projeto, mas apenas os ficheiros de código fonte (`.c/.h` ou `.cpp/.hpp`).

O trabalho deverá ser obrigatoriamente remetido ao docente via *link* de submissão disponível na página Moodle desta UC, até à data e hora definida (também disponível na página Moodle da UC).

A entrega do trabalho prático deverá respeitar **obrigatoriamente** os seguintes requisitos:

- Os ficheiros com o código fonte deverão ser colocados num ficheiro zip com o nome “**VC-GY-xxxx-xxxx-xxxx-xxxx.zip**”, em que:
  - **xxxx** deverá ser preenchido com o número de aluno de cada um dos elementos do grupo);
  - **GY** deverá ser preenchido com **G1** para grupo 1, **G2** para grupo 2, etc.
- Apenas 1 (um) elemento de cada grupo deverá submeter o trabalho.
- No nome do ficheiro zip só deverão constar os números dos alunos que efetivamente realizaram o trabalho, e que, portanto, se irão submeter à avaliação.

O prazo de entrega termina no dia **19 de maio**, às **23:00**. **Não serão considerados trabalhos entregues após esta data.**

## Conduta ética

A falta de transparência em avaliações, presenciais ou não, é naturalmente ilegal e imoral. Todas as fontes utilizadas para suporte a trabalhos devem ser obrigatoriamente e claramente referenciadas. Qualquer plágio, cópia ou conduta académica imprópria será penalizada com a anulação do trabalho. Caso se verifique a existência de trabalhos notoriamente similares (onde por exemplo se tenha alterado apenas os nomes das variáveis de um outro código) entre grupos, todos os trabalhos similares serão anulados.