

# Computação Visual – WebGL

André Santos, 84816 – andrembs@ua.pt

**Resumo** - O presente relatório apresenta uma aplicação desenvolvida em OpenCV e C++ no âmbito da unidade curricular de Computação Visual, de forma a demonstrar algumas das funcionalidades desta tecnologia.

Para isso, a aplicação proposta e desenvolvida é uma detector e avaliador de uma mão de Poker (composta por 7 cartas), em que, a partir de uma imagem que contenha 7 cartas, estas são detectadas e identificadas, e posteriormente são processadas de forma a avaliar essa mesma mão de acordo com as regras do Poker.

**Abstract** - This report presents an application developed in OpenCV and C++ for the Visual Computing course, in order to show some of the functionalities of those technologies.

With that in mind, the proposed and developed application is a 7 card Poker hand detector and evaluator, in which, from a provided image containing 7 cards, those are detected and identified and then processed in order to evaluate that hand accordingly to the Poker rules.

## I. INTRODUÇÃO

Para demonstrar algumas das funcionalidades e capacidades do OpenCV, foi proposto desenvolver uma aplicação de processamento de imagem em C++. Por essa razão, a aplicação escolhida foi um detetor e avaliador de mãos de Poker (compostas por 7 cartas). Estas são identificadas e, de seguida, processadas para que sejam avaliadas de acordo com as regras de Poker.



**Fig. 1 - Mão de Poker**

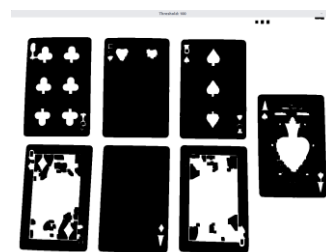
```
Detected cards: Qh Ad Qd As 3s 5h 6c
Hand evaluation: Two Pair, Aces and Queens
```

**Fig. 2 - Resultado da identificação e avaliação**

## II. PRÉ-PROCESSAMENTO

Para que a aplicação possa funcionar corretamente, é necessário utilizar imagens que contenham apenas 7 cartas e que não estejam sobrepostas. De seguida, essa imagem é passada à aplicação no primeiro argumento na execução da mesma.

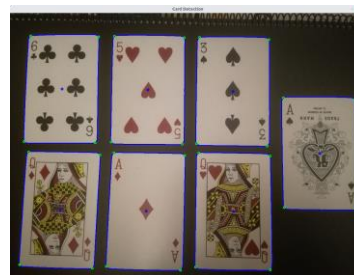
Em primeiro lugar, a imagem a cores é convertida para uma em escala de cinzentos. Depois, é aplicado um thresholding e um closing para que se possa encontrar a forma das 7 cartas.



**Fig. 3 - Thresholding e closing**

A partir dessa imagem, a preto e branco, é utilizada a função `cv::findContours()` para detectar os contornos das cartas e assim, ter a localização das mesmas.

Utilizando os contornos encontrados, as imagens das cartas são recortadas da imagem original.



**Fig. 4 – Detecção de cartas e seus contornos**

## III. IDENTIFICAÇÃO DE VALOR E NAÍPE

A cada uma das imagens recortadas, é encontrado, recortado e dividido ao meio o seu canto superior esquerdo para que se possa descobrir qual é o valor e o naipe da mesma.

Para isso, a cada uma dessas metades do canto é aplicada a mesma função referida anteriormente, obtendo assim um conjunto de contornos que, de seguida, é ordenado de forma decrescente de acordo com a sua área, uma vez que apenas é necessário o maior.

Depois, é calculado o rectângulo que melhor se ajusta ao contorno encontrado. Com esse rectângulo é possível recortar apenas o valor e o naipe de cada carta.



**Fig. 5 - Detecção e identificação do valor**

Para que essa identificação possa ser feita corretamente, cada uma das partes é distorcida para que depois possa ser comparada com as imagens de comparação. Depois é feita a subtração de uma com a outra para que se encontre as imagens mais semelhantes. Desta forma é possível descobrir o valor e naipe de cada carta.

#### IV. AVALIAÇÃO DA MÃO DE POKER

Para a avaliação, foi utilizado um módulo já existente[1]. Assim apenas é necessário passar as abreviaturas do valor e naipe (“9c” para “Nine of Clubs” ou nove de paus, por exemplo) à função `phevaluator::EvaluateCards()`.

Desta forma, conseguimos obter uma descrição da mão de Poker.

#### REFERENCES

- [1] <https://github.com/HenryRLee/PokerHandEvaluator>