

Hearthstone – Duel Status Recognition

André Cardoso, 65069, marquescardoso@ua.pt

Jorge Faustino, 64441, jorgebalseiro@ua.pt

Resumo – Este artigo apresenta uma possível solução para encontrar o vitorioso e um derrotado num jogo de cartas de coleção online. Começa por explicar o jogo em questão e a necessidade de haver um método onde terceiros possam saber qual foi o jogador que ganhou e o que perdeu, assim como alguns detalhes relevantes durante o duelo de cartas.

É abordado as técnicas e métodos usados nas capturas de ecrãs do jogo de maneira a obter as informações relevantes para um eventual sistema de informação de gestão de torneios.

I. INTRODUÇÃO

No âmbito da Unidade Curricular Computação Visual do 4º ano do curso Mestrado Integrado em Engenharia de Computadores e Telemática escolhemos a realização de um projeto usando as bibliotecas de OpenCV[1], o editor Visual Studio 2017[2] utilizando a linguagem de programação c++[3].

O contexto do nosso trabalho está inserido um jogo online, Hearthstone[4], que conta com mais de 70 milhões de jogadores até meio de 2017[5]. No jogo existe uma coleção de 2407 cartas, entre elas, cartas que representam feitiços e monstros com um determinado poder de ataque e defesa que o jogador vai acumulando. Existem também 8 classes, isto é, Mágico, Guerreiro, Caçador e outros que o jogador escolhe como a sua identidade durante o duelo[6]. Os procedimentos para executar um duelo são simples:

1. Cada jogador escolhe uma classe;
2. Cada jogador escolhe um baralho previamente feito com as cartas que possuiu na sua coleção;
3. É feito um duelo até a classe de um jogador ficar sem vida.

Mais pormenores são desnecessários para o contexto deste trabalho. Agora problema encontra-se em saber com um grau de certeza elevado quem foi o vencedor do duelo. O jogo ocorre numa plataforma chegada desenvolvida pela Blizzard[7] onde não existe qualquer tipo de API para sabermos quem ganhou. E como os jogos ocorrem online, onde cada jogador se encontra em espaços físicos completamente distintos, não é viável uma deslocação física ao local para apurar o vencedor.

Uma possível solução é pedir uma capture de ecrã no final do duelo de cada jogador, serem submetidos num serviço e assim calcular com auxílio as ferramentas de análise de imagem do OpenCV quem é o vencedor, o derrotado e a classes que cada jogador usou no seu duelo. A figuras 1 e 2 abaixo representação dois exemplos de uma possível captura de tela feita no final do duelo.



Figure 1- Classe guerreiro, vitória.



Figure 2- Classe Mágico, derrota

Figura 1 representa a classe Guerreiro que venceu o duelo e a figura 2 representa a classe Mágico que perdeu o duelo.

II. IMPLEMENTAÇÃO

A. Filtro de Canny

O primeiro passo foi fazer um levantamento das imagens isoladas de cada classe que designamos por ícone e dos estados possíveis de cada duelo, vitória ou derrota. Como estas imagens serão usadas repetidamente durante as comparações, foi gerado e guardado em disco a sua representação aplicado o Filtro de Canny com um *threshold* de 86, como podemos verificar na Figura 3 e Figura 4.



Figure 1- Ícone guerreiro com e sem de Filtro Canny.

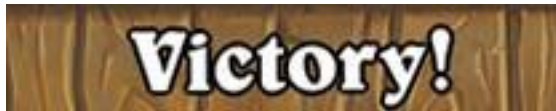


Figure 4- Resultado de vitória com e sem o Filtro de Canny.

Para obter a melhor solução usamos os métodos e técnicas de *Template Matching*. A ordem de execução para descobrir a classe representada e o resultado do duelo na captura de ecrã passa, num primeiro passo, por fazer o redimensionamento da imagem para uma altura de 720 pixéis e a largura com o tamanho mais adequado para manter o rácio da imagem.

De seguida aplicamos o *Filtro de Canny* na captura de ecrã, como utilizado nos ícones, representado na figura 3 e é executado uma comparação da captura de ecrã por todos os ícones que existem na nossa biblioteca de possibilidades. Uma nova matriz é gerada com as duas imagens envolvidas, onde encontramos uma métrica que nos indica o seu grau de semelhança, quando maior o seu valor mais parecido o do ícone que estamos a comparar.

B. Pontuação final

Apenas aplicando o método de filtragem do Filtro de Canny, verificamos que não chegava uma vez que tínhamos uma taxa de sucesso de 50% das imagens que usamos para desenvolvimento da solução. Para minimizar os falsos positivos encontrados, desenvolvemos uma métrica que tem em conta para além do grau de semelhança, a posição onde se encontra esse grau. Isto porque, dentro do nosso contexto todas as nossas áreas de interesse passam pelo centro da imagem.

Para isso as fórmulas de calculo da pontuação final, usamos a distância do centro da área onde o ícone tem maior semelhança com o centro da captura de ecrã, subtraído com o valor do pixel com maior semelhança, multiplicando pela normalização. Como o valor da maior semelhança está compreendido entre 0 e 1, a normalização vem aumentar a sua relevância no resultado.

$$\text{Pontuação} = \text{distancia} - \text{match_value} * 80$$

III. RESULTADO FINAL

A execução da nossa solução final para uma captura de ecrã da classe Guerreiro com o resultado de vitória apresenta os valores representados na Tabela 1. Quanto menor for o valor mais próximo da realidade se encontra.

Ícone	Pontuação
Druid	367.894
Hunter	294.722
Mage	120.871
Paladin	243.534
Priest	321.321
Rogue	341.536
Warlock	301.509
Warrior	3.12035
Shaman	601.735
Victory	-32.265
Defeat	15.465

Tabela 1- Pontuação final da Figura 1

Os valores apresentados na tabela anterior são os da Figura 1, apresentada no capítulo de introdução.

A execução para a captura de ecrã da classe Mágico com o resultado de derrota apresenta os valores representados na Tabela 2. Assim como a tabela anterior, quando menor for o valor mais próximo da realidade se encontra.

Ícone	Pontuação
Druid	292.196
Hunter	280.289
Mage	11.66
Paladin	348.25
Priest	227.916
Rogue	309.146
Warlock	170.757
Warrior	260.309
Shaman	602.152
Victory	256.66
Defeat	-14.8705

Tabela 2- Pontuação final da Figura 2

Como podemos verificar, pontuação final não deixa dúvidas para qual é classe e o resultado na captura de ecrã. Figura 5 apresenta a imagem durante o processo de calculo de pontuação.



Figure 5- Representação intermediária da captura durante cálculo do resultado final.

Os quadrados a preto representam a localização da melhor semelhança calculada para as outras possíveis classes e dos possíveis resultados.

IV. CONCLUSÃO

Verificamos que análise de imagem é em si complexa e com muita utilidade em vários campos, para compreender certos objetos e padrões encontrados nas mais variadas áreas de trabalho científico e de engenharia.

Consideramos que este modulo desenvolvimento possa ser usado mais tarde por uma plataforma que gere torneios ou duelos entre vários utilizadores de maneira a definir quem foi o vencedor sem intervenção humana.

Este trabalho foi muito importante para o nosso conhecimento, o aprofundamento deste tema fez-nos aprender e compreender o método e técnicas utilizadas na manipulação de imagens e vídeos. Cumprimos todos os objetivos que nos foram propostos neste projeto.

V. REFERENCES

- [1] "OpenCV." [Online]. Available: <https://opencv.org/>. [Accessed: 01-Jan-2019].
- [2] "Visual Studio 2017." [Online]. Available: <https://visualstudio.microsoft.com/downloads/>. [Accessed: 01-Jan-2019].
- [3] "C++." [Online]. Available: <http://www.cplusplus.com/doc/tutorial/>. [Accessed: 01-Jan-2019].
- [4] "HeathStone." [Online]. Available: <https://playhearthstone.com/en-us/>. [Accessed: 01-Jan-2019].
- [5] "Player Count." [Online]. Available: <https://www.pcgamesn.com/hearthstone/hearthstone-player-count>. [Accessed: 01-Jan-2019].
- [6] "Game Cards." [Online]. Available: https://hearthstone.gamepedia.com/Card#Collecting_cards. [Accessed: 01-Jan-2019].
- [7] "Blizzard." [Online]. Available: <https://www.blizzard.com/en-us/>. [Accessed: 01-Jan-2019].