****

Projet annuel

Jeu de plateau en réalité augmentée

Rapport

à

Mi-parcours

Etudiants:

*Alban Morin*

*Charaf-Eddine Skiker*

**Contexte:**

Les joueurs cherchent de plus en plus de jouer en multi-joueurs les jeux de plateaux comme les échecs ou le jeu de dames, ils préfèrent ainsi jouer sur des plateaux réels et souhaitent pouvoir jouer sur un plateau réel contre une Intelligence artificielle.

**Objectif:**

Développer une application permettant de jouer sur un plateau physique en réalité augmentée. Contre une intelligence artificielle, ou un autre joueur.

**OpenCV:**

OpenCV - Open [Computer Vision](https://fr.wikipedia.org/wiki/Vision_par_ordinateur) - est une [bibliothèque graphique](https://fr.wikipedia.org/wiki/Biblioth%C3%A8que_graphique) [libre](https://fr.wikipedia.org/wiki/Logiciel_libre), initialement développée par [Intel](https://fr.wikipedia.org/wiki/Intel), spécialisée dans le [traitement d'images](https://fr.wikipedia.org/wiki/Traitement_d%27images) en temps réel.

OpenCV est la librairie principale utilisé dans ce projet, d’abord pour déterminer et reconnaitre un tableau, et puis pour gérer les actions du jeu en reconnaissant les pions et leurs mouvements.

**Partie I : Lecture des QR-Codes**

La lecture des QR codes nous permette de faciliter la lecture du tableau et ses cases, Chaque case est caractérisé par un QR code qui déterminera sa position ainsi que le pion posé dessus.

1. Démarrer un streaming en camera web :

-Utilisation de la librairie **sarxos** **webcam** : permettre aux webcams intégrées ou connectées via USB d'accéder directement à partir de Java, les utilisateurs peuvent lire les images directement de leur caméra.

-Générer un panel qui affiche une lecture vidéo en direct.

2. Lecture du QR code :

-utilisation de la librairie **ZXing :** -zebra crossing- est une open-source librairie, à la base pour décoder des multi-formats 1D/2D barcodes, implémenté en Java.

- capturer est décoder le QR code scanné mit face la webcam :

result = **new** MultiFormatReader().decode(bitmap);

-

-utilisation des threads pour pouvoir continuer à lire encore des QR Codes.

**Partie II : Calibration du plateau**

OpenCV permet de calibrer une image suivant la lecture de ses coins aussi que les lignes constituant l’image. Dans ce cas on a utilisé une calibration en se basant sur les coins et les lignes verticaux et horizontaux d’un tableau. (Echiquier de 9x6 pour le test).

//classes utilisées

//explications methodes

//Utilisation de JavaFX pour l’affichage xml (librairie xml)

//ligne de codes