Mathable Game Reader

Bejenariu David-Cosmin, 407

I. Introduction

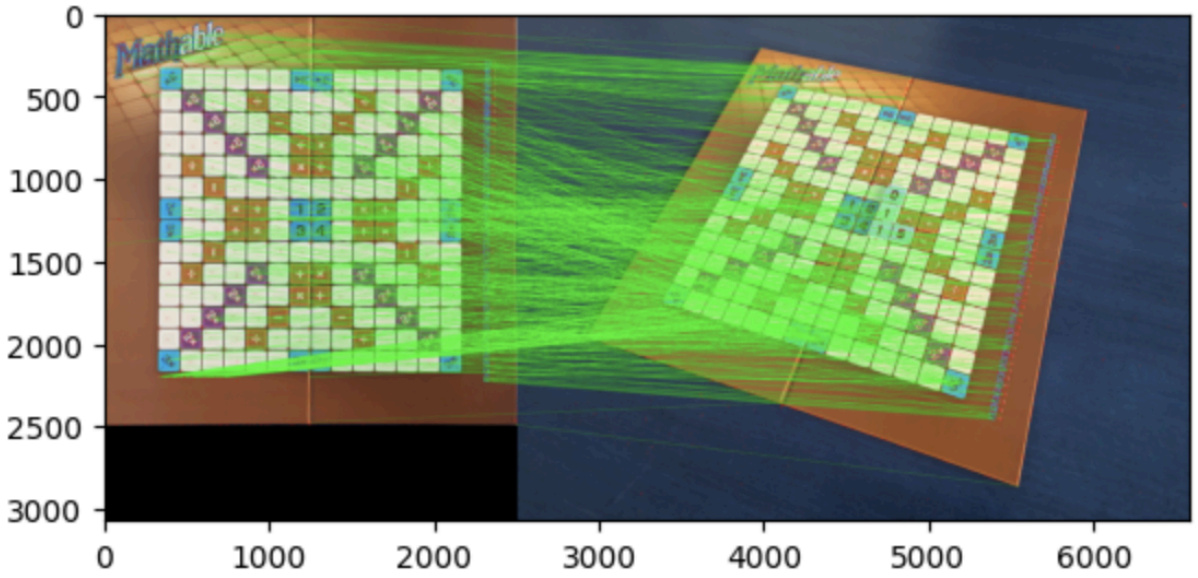
Mathable Game Reader is the first task of the Computer Vision course. Given a training set of 200 images depicting 4 games of 50 turns of Mathable, students would provide an algorithm to process the board configuration and calculate scores for other 4 games of 50 testing turns.

The task consists of detecting and recognizing new tokens placed on the board, as well as calculating the score for 1 of the 2 players performing the current turn. The implementation steps are explained in the following sections.

II. Board Aquisition

The first step is the board acquisition. For this subtask, I used one of the model boards given in the *board+tokens* directory, which I manually cropped and corner-adjusted to match a top-down perspective.

With the model board set, I computed keypoints and descriptors for the other images using SIFT. The keypoints were then matched between each test image and the model image using the KNN matching algorithm of the FlannBasedMatcher. Best matching keypoints were then fed into a homography matrix which then helped compute the perspective warp of the test image.



III. Token Detection

Histogram of hues comparison was used to detect whether the position of a tile has changed since the last configuration. For that, each turn the algorithm splits the image board into a 14x14 matrix to get the patches. For each patch, the histogram of hues is computed and compared with the same position on the previous board configuration, by calculating their intersection. The smallest intersection of hue histograms means that the patches are quite different, thus a token has been placed there.

IV. Token Recognition

For the token recognition I used the tokens extracted from the train images, along with the additional tokens from the *board+tokens* directory. HOG descriptors were then extracted from the images and fed into a Linear SVM classifier. For each test image, the algorithm would extract the new patch from the board and predict its token using the trained model.