

Mise en place de l'environnement d'entraînement de notre IA sur les pièces d'échecs

SAE_501

Version python : [3.12](#)

<https://docs.ultralytics.com/modes/train/#introduction>

J'ai choisi ce dataset étant donné qu'il était plus complet que le précédent :

<https://www.kaggle.com/datasets/kirithiik/chess-pieces/data>

Au début, je comptais réaliser les annotations des images via labelImg, qui est un outil d'annotation graphique, mais j'ai rencontré quelques problèmes et étant que j'avais déjà repéré d'autres alternatives, j'ai décidé de me diriger vers **Roboflow**.

Roboflow est en soit un outil beaucoup plus lourd que labelImg, car il permet aussi d'accéder à beaucoup de données sous forme de graphique par exemple, mais surtout d'assigner un certain nombre d'image à chaque membre du projet, ce qui se montrera sans doute utile plus tard lorsque nous décidons d'ajouter plus d'image pour la version final de notre IA.

Dataset Versions

Create New Version

VERSIONS

2024-11-01 1:34pm
v2 · a month ago
299 640x640
Stretch to

2024-11-01 12:37pm
v1 · a month ago
299 Fast COCO

v2 2024-11-01 1:34pm
Generated on Nov 1, 2024

Download Dataset Edit

This version doesn't have a model.

Train an optimized, state of the art model with Roboflow or upload a custom trained model to use features like Label Assist and Model Evaluation and deployment options like our auto-scaling API and edge device support.

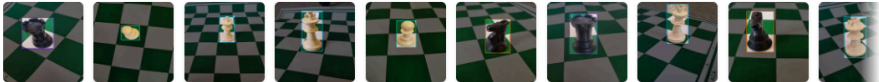
Custom Train and Upload

Train with Roboflow

Available Credits: 2

299 Total Images

View All Images →



Dataset Split

TRAIN SET
209 Images
70%

VALID SET
60 Images
20%

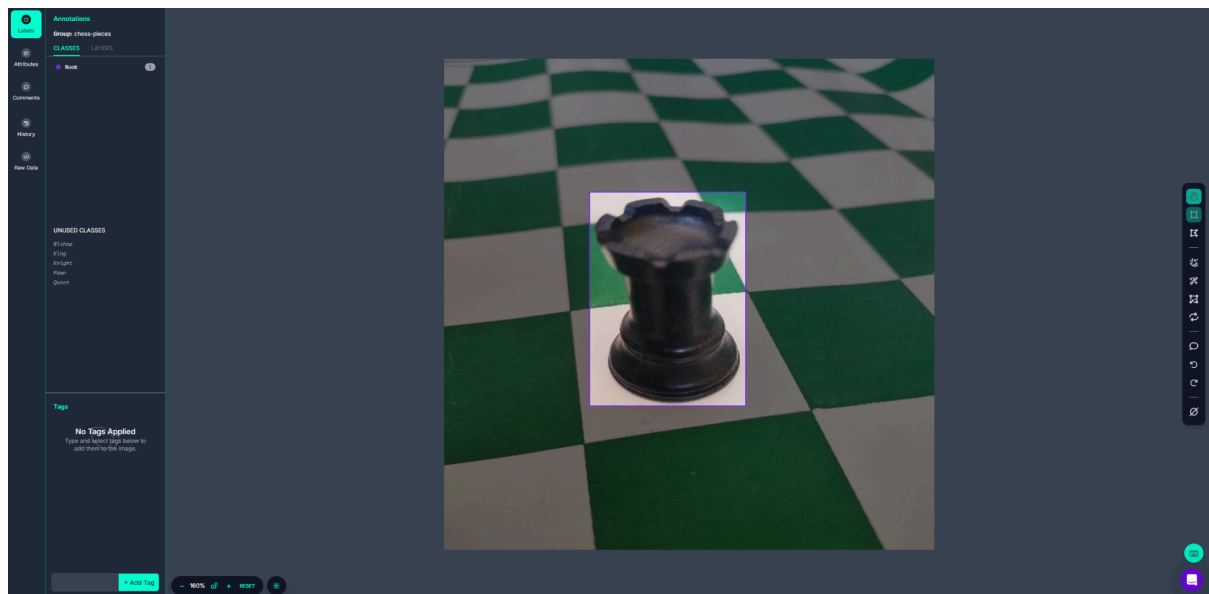
TEST SET
30 Images
10%

Preprocessing

Auto-Orient: Applied
Resize: Stretch to 640x640

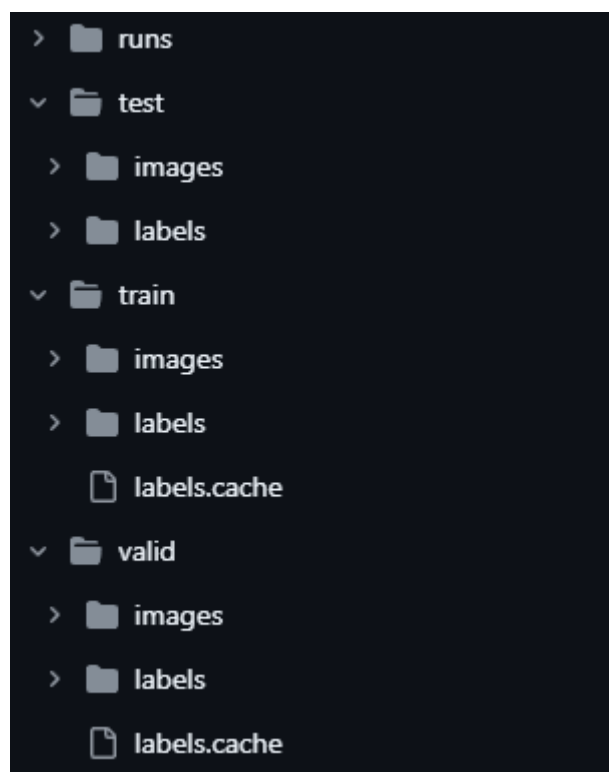
Augmentations

No augmentations were applied.



J'ai importé mon Dataset sur roboflow et ai commencé à mettre des Labels sur nos images pour la reconnaissance.

Il faut maintenant annoter les images, une tâche qui consiste à isoler ce que l'on veut détecter, dans notre cas les différentes pièces d'échec (Roi, Reine, Tour, Fou, Cavalier et Pion). Ensuite, Roboflow nous soulage de la tâche qu'est la répartition des données :



Dans notre cas, 70% des images sont mises dans le dossier Train, images qui serviront donc à entraîner le modèle sur nos données, 20% sont mises dans validation, la validation sert à éviter les choses comme le sur apprentissage (éviter que le modèle mémorise juste les données d'entraînement). Et les 20% restant finissent dans test.

Grossièrement, on pourrait dire que les données d'entraînement sont les exercices d'apprentissage, celles de validation sont les exercices pour affiner nos connaissances et celles de test représenterait l'examen final.

Une fois le test lancé, les résultats seront envoyés dans le dossier /runs, et on peut ainsi utiliser le fichier créé dans notre application directement pour détecter nos images.