

Projet d'Intelligence Artificielle

Thibaud D.

Système de reconnaissance faciale

Ce projet d'intelligence artificielle a été réalisé dans le cadre d'un projet académique en IA et Deep Learning, en binôme. L'objectif principal était de concevoir un système de reconnaissance faciale capable d'identifier des visages à partir d'images, en s'appuyant sur des réseaux de neurones convolutifs et la bibliothèque PyTorch.

Le projet repose sur l'ensemble du pipeline de reconnaissance faciale, depuis la préparation du jeu de données jusqu'à l'inférence du modèle entraîné. Une attention particulière a été portée à l'organisation et à la qualité des données. Des scripts dédiés ont été développés afin d'automatiser la séparation des images entre les ensembles d'entraînement et de validation, ainsi que pour vérifier la cohérence et la propreté du dataset.

Plusieurs architectures de réseaux de neurones ont été étudiées et testées dans une démarche de recherche et d'optimisation des performances. Dans ce cadre, un travail a été mené sur les modèles ConvNeXt et EfficientNet-B3, incluant leur entraînement, l'ajustement des paramètres et l'évaluation des résultats. Les modèles entraînés ont ensuite été intégrés dans un script d'inférence (`student_infer.py`), permettant de tester automatiquement la reconnaissance faciale sur de nouvelles images.

Le système final est compatible avec un script d'évaluation externe imposé, garantissant une exécution contrôlée, des temps de réponse limités et une évaluation objective des performances. Ce projet a permis de mettre en pratique des notions avancées telles que :

- Le Deep Learning appliqué à la vision par ordinateur
- La gestion et la préparation de datasets d'images
- L'entraînement et l'évaluation de modèles CNN
- Ainsi que le travail collaboratif et la répartition des tâches dans un projet de recherche.

Ce projet illustre une approche complète et rigoureuse du développement d'un système d'intelligence artificielle, depuis la phase de recherche jusqu'à l'intégration et aux tests finaux.

7-7.jpg	true= 7 pred= 7 acc=0.9277 [0.446s]
71-7.jpg	true= 71 pred= 71 acc=0.9286 [0.339s]
72-7.jpg	true= 72 pred= 29 acc=0.9176 [0.401s]
73-7.jpg	true= 73 pred= 73 acc=0.9186 [0.348s]
74-7.jpg	true= 74 pred= 74 acc=0.9195 [0.337s]
75-7.jpg	true= 75 pred= 75 acc=0.9285 [0.360s]
76-7.jpg	true= 76 pred= 76 acc=0.9213 [0.349s]
77-7.jpg	true= 77 pred= 77 acc=0.9222 [0.415s]
78-7.jpg	true= 78 pred= 78 acc=0.9231 [0.467s]
79-7.jpg	true= 79 pred= 79 acc=0.9239 [0.431s]
8-7.jpg	true= 8 pred= 8 acc=0.9247 [0.386s]
80-7.jpg	true= 80 pred= 80 acc=0.9255 [0.428s]
81-7.jpg	true= 81 pred= 81 acc=0.9263 [0.345s]
82-7.jpg	true= 82 pred= 82 acc=0.9271 [0.389s]
83-7.JPG	true= 83 pred= 83 acc=0.9278 [0.364s]
84-7.jpg	true= 84 pred= 84 acc=0.9286 [0.376s]
85-7.jpg	true= 85 pred= 85 acc=0.9293 [0.313s]
86-7.jpg	true= 86 pred= 6 acc=0.9280 [0.349s]
87-7.jpg	true= 87 pred= 87 acc=0.9288 [0.393s]
88-7.jpg	true= 88 pred= 88 acc=0.9216 [0.410s]
89-7.jpg	true= 89 pred= 89 acc=0.9223 [0.330s]
9-7.jpg	true= 9 pred= 36 acc=0.9135 [0.318s]
90-7.JPG	true= 90 pred= 90 acc=0.9143 [0.593s]
91-7.jpg	true= 91 pred= 91 acc=0.9151 [0.403s]
92-7.jpg	true= 92 pred= 92 acc=0.9159 [0.392s]
93-7.jpg	true= 93 pred= 93 acc=0.9167 [0.426s]
94-7.jpg	true= 94 pred= 94 acc=0.9174 [0.317s]
95-7.jpg	true= 95 pred= 95 acc=0.9182 [0.362s]
96-7.jpg	true= 96 pred= 96 acc=0.9189 [0.407s]
97-7.JPG	true= 97 pred= 97 acc=0.9196 [0.395s]
98-7.jpg	true= 98 pred= 98 acc=0.9284 [0.474s]
99-7.jpg	true= 99 pred= 99 acc=0.9211 [0.607s]

Evaluated: 114 Correct: 105 Skipped: 0 Final accuracy: 0.9211
Average per-Image time: 0.384s | Throughput: 2.60 img/s

Voici un exemple de sortie présenté ci-dessus. Lors de la soutenance, le professeur a utilisé le système en lui fournissant des photos avec lesquelles il n'avait pas été entraîné. Le modèle a alors atteint un taux de réussite de 83 %.

Le jeu de données utilisé pour l'entraînement du modèle (ensembles d'entraînement et de validation confondus) était composé de 798 images.



APPROUVÉ



NON RECONNNU

