

Projet Deep Learning MAPI3 2021-2022

Abdellah ABARDAME
Dani HALEGUA
Luca CHAMPESTING
Idyano LEROY

Soutenu le :

10 mars 2022

Introduction

Le dataset utilisé dans ce projet contient 4000 images, appartenant à 7 catégories.



Figure – Quelques exemples d'images du dataset.

Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 254, 254, 16)	448
max_pooling2d (MaxPooling2D)	(None, 127, 127, 16)	0
conv2d_1 (Conv2D)	(None, 125, 125, 32)	4640
max_pooling2d_1 (MaxPooling2D)	(None, 62, 62, 32)	0
conv2d_2 (Conv2D)	(None, 60, 60, 32)	9248
flatten (Flatten)	(None, 115200)	0
dense (Dense)	(None, 32)	3686432
dense_1 (Dense)	(None, 7)	231
=====		
Total params: 3,700,999		
Trainable params: 3,700,999		
Non-trainable params: 0		

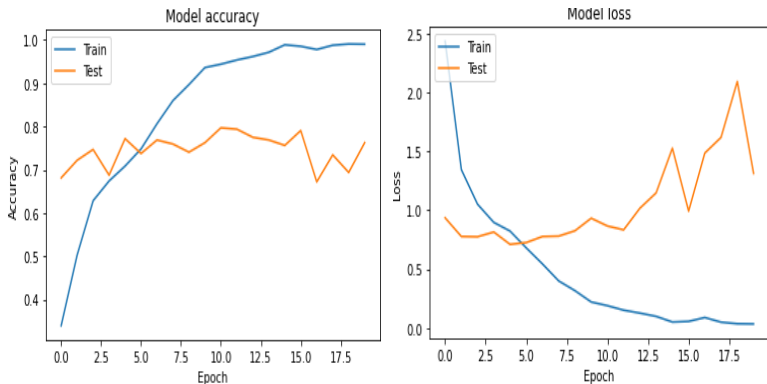


Figure – Courbes d'apprentissage du CNN.

Méthodes pour limiter l'over-fitting

- Data augmentation
- Batch normalisation
- Dropout layers + régularisation

Data augmentation

- Augmente la taille des données d'entraînement
- Ajoute de la variance dans le jeu d'entraînement
- Transformations qui n'affecte pas la classe de l'image



Figure – Exemple de data augmentation sur une image.

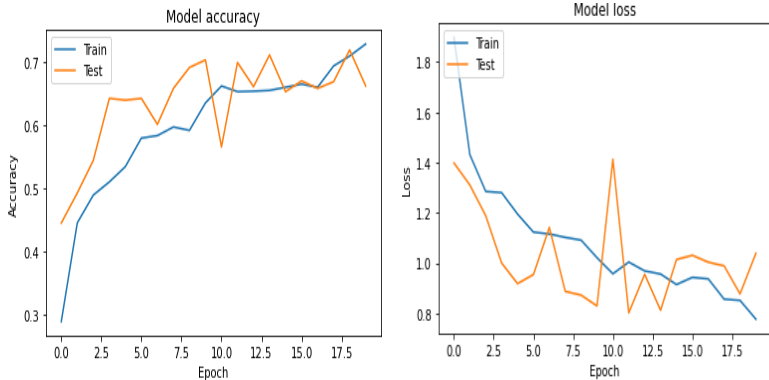


Figure – Courbes d'apprentissage du CNN avec data augmentation.

Batch normalisation

- Normalise les données à l'intérieur du réseau
- Peut améliorer les performances du modèle
- Accélère le processus d'entraînement

Layer (type)	Output Shape	Param #
conv2d_18 (Conv2D)	(None, 254, 254, 16)	448
batch_normalization_12 (Batch Normalization)	(None, 254, 254, 16)	64
max_pooling2d_12 (MaxPooling2D)	(None, 127, 127, 16)	0
conv2d_19 (Conv2D)	(None, 125, 125, 32)	4640
batch_normalization_13 (Batch Normalization)	(None, 125, 125, 32)	128
max_pooling2d_13 (MaxPooling2D)	(None, 62, 62, 32)	0
conv2d_20 (Conv2D)	(None, 60, 60, 32)	9248
batch_normalization_14 (Batch Normalization)	(None, 60, 60, 32)	128
flatten_6 (Flatten)	(None, 115200)	0
dense_12 (Dense)	(None, 32)	3686432
batch_normalization_15 (Batch Normalization)	(None, 32)	128
dense_13 (Dense)	(None, 7)	231
Total params: 3,701,447		
Trainable params: 3,701,223		
Non-trainable params: 224		

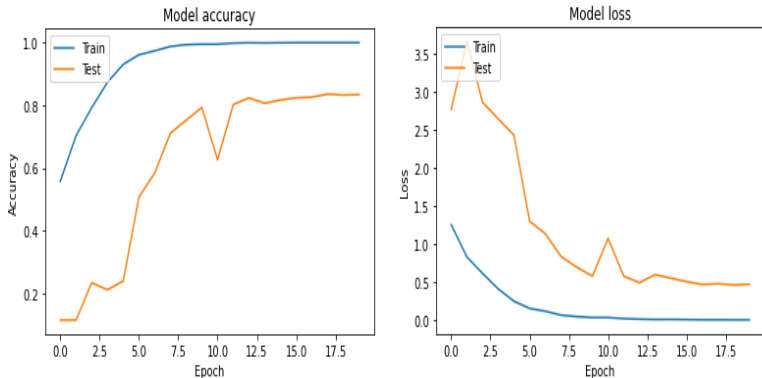


Figure – Courbes d'apprentissage du CNN avec batch normalisation.

Dropout layers + régularisation

- Empêche le réseau d'apprendre des modèles trop spécifique
- Supprime aléatoirement une portion des entrées d'une couche
- Force le réseau à rechercher des modèles plus généraux.

Dropout layers + régularisation

Layer (type)	Output Shape	Param #
dropout_30 (Dropout)	(None, 256, 256, 3)	0
conv2d_40 (Conv2D)	(None, 254, 254, 16)	448
dropout_31 (Dropout)	(None, 254, 254, 16)	0
max_pooling2d_26 (MaxPoolin g2D)	(None, 127, 127, 16)	0
conv2d_41 (Conv2D)	(None, 125, 125, 16)	2320
dropout_32 (Dropout)	(None, 125, 125, 16)	0
max_pooling2d_27 (MaxPoolin g2D)	(None, 62, 62, 16)	0
conv2d_42 (Conv2D)	(None, 60, 60, 16)	2320
dropout_33 (Dropout)	(None, 60, 60, 16)	0
flatten_12 (Flatten)	(None, 57600)	0
dense_24 (Dense)	(None, 32)	1843232
dropout_34 (Dropout)	(None, 32)	0
dense_25 (Dense)	(None, 7)	231
=====		
Total params: 1,848,551		
Trainable params: 1,848,551		
Non-trainable params: 0		

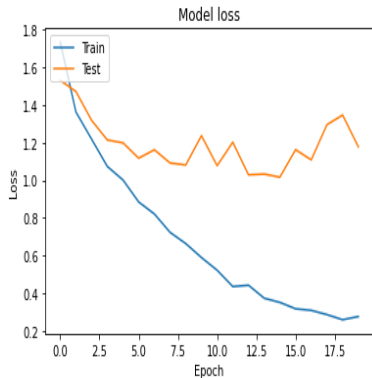
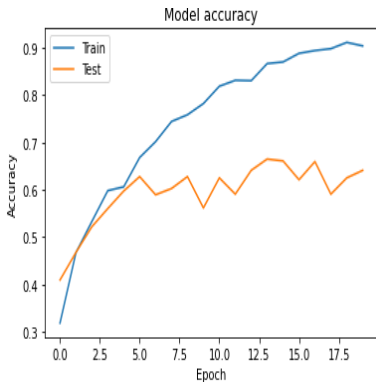


Figure – Courbes d'apprentissage du CNN avec dropout layers.

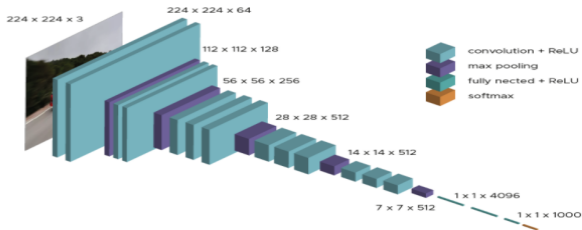
Transfert learning

- Récupération d'un modèle pré-entraîné sur une tâche similaire
- Geler les couches du modèle de base, pour pas perdre les informations obtenues
- Rajout d'un petit modèle de classification qu'on entraîne avec les poids du modèle de base

VGG16



Architecture Algorithm VGG16



Architecture du réseau VGG16.

Transfert learning

Layer (type)	Output Shape	Param #
vgg16 (Functional)	(None, 8, 8, 512)	14714688
flatten_17 (Flatten)	(None, 32768)	0
dense_39 (Dense)	(None, 256)	8388864
dense_40 (Dense)	(None, 7)	1799
=====		
Total params: 23,105,351		
Trainable params: 8,390,663		
Non-trainable params: 14,714,688		

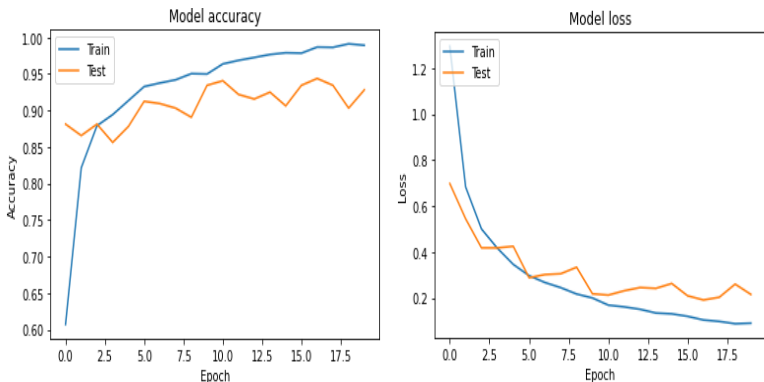


Figure – Courbes d'apprentissage du réseau VGG16.

Résultats

Méthode utilisée :	train loss	train accuracy	test loss	test accuracy
CNN basique	0.0348	0.9902	1.3145	0.7625
Data augmentation	0.7776	0.7285	1.0394	0.6623
Batch normalization	0.0027	1.0000	0.4704	0.8344
Dropout layers	0.2777	0.9041	1.1777	0.6411
Transfert learning avec VGG16	0.0908	0.9892	0.2155	0.9281

Figure – Résultats des différents CNN.

Évaluation du modèle

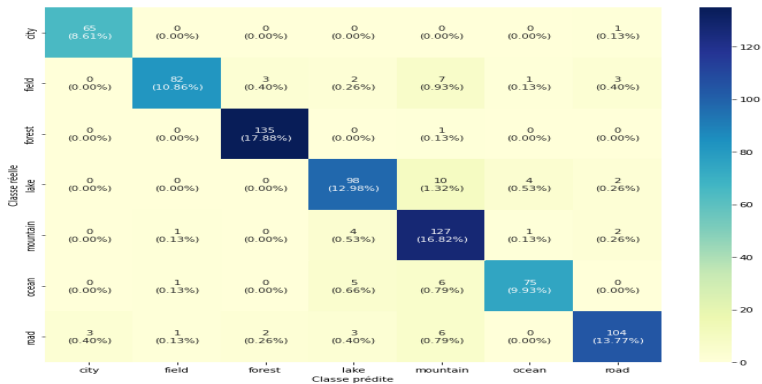


Figure – Matrice de confusion des prédictions.

Mauvaises prédictions

Mauvaises prédictions

Véritable classe: field
Classe prédite: road



Véritable classe: lake
Classe prédite: forest



Véritable classe: field
Classe prédite: lake



Véritable classe: mountain
Classe prédite: road



Véritable classe: mountain
Classe prédite: ocean



Véritable classe: road
Classe prédite: mountain



Véritable classe: mountain
Classe prédite: field



Véritable classe: road
Classe prédite: mountain



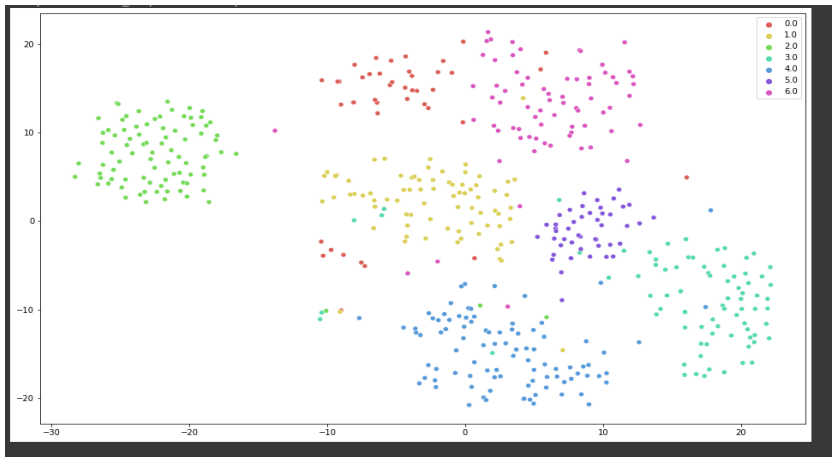
Véritable classe: city
Classe prédite: road



Véritable classe: road
Classe prédite: mountain



t-SNE



3 plus proches voisins

