

Deep Learning par la Pratique

Implémentation d'un CNN avec Keras

Définition d'un modèle CNN

```
1 model = Models.Sequential()
2 #Un premier Layer de 10 convolutions de 3x3 pixels
3 model.add(Layers.Conv2D(10,kernel_size=(3,3),activation='relu',
4                           input_shape=(150,150,3)))
5 #Un Layer "max pooling"
6 model.add(Layers.MaxPool2D(3,3))
7 #Un Layer "Flatten"
8 model.add(Layers.Flatten())
9 #Un Layer "Dense" avec 6 sorties et un softmax
10 model.add(Layers.Dense(6,activation='softmax'))
11 #Compilation du modèle avec la définition de la loss
12 model.compile(optimizer=Optimizer.Adam(lr=0.0001),
13               loss='sparse_categorical_crossentropy',
14               metrics=['accuracy'])
```

Affichage du modèle

```
1 model.summary()
```

```
1 -----
2 Layer (type)                Output Shape                Param #
3 -----
4 conv2d (Conv2D)             (None, 148, 148, 10)        280
5 -----
6 max_pooling2d (MaxPooling2D) (None, 49, 49, 10)          0
7 -----
8 flatten (Flatten)           (None, 24010)                0
9 -----
10 dense (Dense)               (None, 6)                    144066
11 =====
12 Total params: 144,346
13 Trainable params: 144,346
14 Non-trainable params: 0
15 -----
```