

# CNN – Classification Dogs vs Cats

Objectif : entraîner un CNN pour une classification binaire et analyser l'impact du nombre de convolutions.

```
In [42]: import numpy as np
import matplotlib.pyplot as plt
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Flatten, Dense
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.callbacks import EarlyStopping, ModelCheckpoint, Callback
import os
import time
import GPUtil
import psutil
```

```
In [43]: IMG_SIZE = (64, 64)
BATCH_SIZE = 32
EPOCHS = 20
```

```
In [44]: train_gen = ImageDataGenerator(
    rescale=1./255,
    zoom_range=0.2,
    shear_range=0.2,
    horizontal_flip=True
)

test_gen = ImageDataGenerator(rescale=1./255)

train_set = train_gen.flow_from_directory(
    "../dataset/training_set",
    target_size=IMG_SIZE,
    batch_size=BATCH_SIZE,
    class_mode="binary"
)

test_set = test_gen.flow_from_directory(
    "../dataset/test_set",
    target_size=IMG_SIZE,
    batch_size=BATCH_SIZE,
    class_mode="binary"
)
```

Found 8000 images belonging to 2 classes.  
Found 2000 images belonging to 2 classes.

```
In [45]: def build_cnn(nb_conv):
    model = Sequential()

    # 1ère convolution
    model.add(Conv2D(32, (3,3), activation='relu', input_shape=(64,64,3)))
    model.add(MaxPooling2D((2,2)))

    # 2ème convolution
    model.add(Conv2D(32, (3,3), activation='relu'))
```

```

model.add(MaxPooling2D((2,2)))

# 3ème convolution OPTIONNELLE
if nb_conv == 3:
    model.add(Conv2D(64, (3,3), activation='relu'))
    model.add(MaxPooling2D((2,2)))

model.add(Flatten())
model.add(Dense(128, activation='relu'))
model.add(Dense(1, activation='sigmoid'))

model.compile(
    optimizer='adam',
    loss='binary_crossentropy',
    metrics=['accuracy']
)

return model

```

```

In [46]: def get_cpu_ram():
    cpu_percent = psutil.cpu_percent(interval=1)
    ram = psutil.virtual_memory()
    ram_used_mb = ram.used / (1024 ** 2)
    return cpu_percent, ram_used_mb
def get_gpu_stats():

    try:
        gpus = GPUtil.getGPUs()
        if not gpus:
            return None, None, None
        gpu = gpus[0]
        return gpu.load * 100, gpu.memoryUsed, gpu.memoryTotal
    except:
        return None, None, None

```

```

In [47]: class PerformanceCallback(Callback):
    def on_epoch_begin(self, epoch, logs=None):
        self.start_time = time.time()

    def on_epoch_end(self, epoch, logs=None):
        cpu, ram = get_cpu_ram()
        gpu, vram_used, vram_total = get_gpu_stats()
        duration = time.time() - self.start_time

        msg = f" | CPU: {cpu:.1f}% | RAM: {ram:.0f} MB | Time: {duration:.1f}s"
        if gpu is not None:
            msg += f" | GPU: {gpu:.1f}% | VRAM: {vram_used}/{vram_total} MB"
        print(msg)

    def get_callbacks(nb_conv):
        ckpt_dir = f"checkpoints_cnn_{nb_conv}"
        os.makedirs(ckpt_dir, exist_ok=True)

        checkpoint = ModelCheckpoint(
            filepath=os.path.join(ckpt_dir, "weights_epoch_{epoch:02d}.h5"),
            monitor="val_loss",
            save_weights_only=True,
            save_best_only=False,

```

```
        verbose=1
    )

    early_stop = EarlyStopping(
        monitor="val_loss",
        patience=3,
        restore_best_weights=True,
        verbose=1
    )

    return checkpoint, early_stop, ckpt_dir
```

```
In [48]: model_2 = build_cnn(nb_conv=2)

checkpoint, early_stop, ckpt_dir = get_callbacks(2)

history_2 = model_2.fit(
    train_set,
    steps_per_epoch=train_set.samples // BATCH_SIZE,
    epochs=20,
    validation_data=test_set,
    validation_steps=test_set.samples // BATCH_SIZE,
    callbacks=[checkpoint, early_stop, PerformanceCallback()],
    verbose=1
)
```

Epoch 1/20  
250/250 [=====] - ETA: 0s - loss: 0.6836 - accuracy: 0.5564  
Epoch 1: saving model to checkpoints\_cnn\_2\weights\_epoch\_01.h5  
| CPU: 4.6% | RAM: 15655 MB | Time: 14.0s | GPU: 30.0% | VRAM: 764.0/16303.0 MB  
250/250 [=====] - 14s 56ms/step - loss: 0.6836 - accuracy: 0.5564 - val\_loss: 0.6417 - val\_accuracy: 0.6028  
Epoch 2/20  
249/250 [=====>.] - ETA: 0s - loss: 0.6052 - accuracy: 0.6711  
Epoch 2: saving model to checkpoints\_cnn\_2\weights\_epoch\_02.h5  
| CPU: 3.6% | RAM: 15688 MB | Time: 12.1s | GPU: 42.0% | VRAM: 764.0/16303.0 MB  
250/250 [=====] - 12s 48ms/step - loss: 0.6051 - accuracy: 0.6712 - val\_loss: 0.5606 - val\_accuracy: 0.7233  
Epoch 3/20  
249/250 [=====>.] - ETA: 0s - loss: 0.5646 - accuracy: 0.7101  
Epoch 3: saving model to checkpoints\_cnn\_2\weights\_epoch\_03.h5  
| CPU: 2.7% | RAM: 15681 MB | Time: 12.1s | GPU: 2.0% | VRAM: 774.0/16303.0 MB  
250/250 [=====] - 12s 48ms/step - loss: 0.5647 - accuracy: 0.7100 - val\_loss: 0.5898 - val\_accuracy: 0.7082  
Epoch 4/20  
249/250 [=====>.] - ETA: 0s - loss: 0.5413 - accuracy: 0.7265  
Epoch 4: saving model to checkpoints\_cnn\_2\weights\_epoch\_04.h5  
| CPU: 4.1% | RAM: 15687 MB | Time: 12.3s | GPU: 34.0% | VRAM: 781.0/16303.0 MB  
250/250 [=====] - 12s 49ms/step - loss: 0.5412 - accuracy: 0.7269 - val\_loss: 0.5411 - val\_accuracy: 0.7349  
Epoch 5/20  
249/250 [=====>.] - ETA: 0s - loss: 0.5200 - accuracy: 0.7390  
Epoch 5: saving model to checkpoints\_cnn\_2\weights\_epoch\_05.h5  
| CPU: 2.3% | RAM: 15684 MB | Time: 11.8s | GPU: 3.0% | VRAM: 781.0/16303.0 MB  
250/250 [=====] - 12s 47ms/step - loss: 0.5201 - accuracy: 0.7391 - val\_loss: 0.5263 - val\_accuracy: 0.7404  
Epoch 6/20  
249/250 [=====>.] - ETA: 0s - loss: 0.4991 - accuracy: 0.7572  
Epoch 6: saving model to checkpoints\_cnn\_2\weights\_epoch\_06.h5  
| CPU: 1.6% | RAM: 15669 MB | Time: 11.7s | GPU: 1.0% | VRAM: 781.0/16303.0 MB  
250/250 [=====] - 12s 47ms/step - loss: 0.4998 - accuracy: 0.7570 - val\_loss: 0.4955 - val\_accuracy: 0.7656  
Epoch 7/20  
249/250 [=====>.] - ETA: 0s - loss: 0.4843 - accuracy: 0.7643  
Epoch 7: saving model to checkpoints\_cnn\_2\weights\_epoch\_07.h5  
| CPU: 2.4% | RAM: 15545 MB | Time: 11.5s | GPU: 2.0% | VRAM: 833.0/16303.0 MB  
250/250 [=====] - 12s 46ms/step - loss: 0.4839 - accuracy: 0.7648 - val\_loss: 0.5282 - val\_accuracy: 0.7535  
Epoch 8/20  
249/250 [=====>.] - ETA: 0s - loss: 0.4732 - accuracy: 0.7713  
Epoch 8: saving model to checkpoints\_cnn\_2\weights\_epoch\_08.h5  
| CPU: 3.4% | RAM: 15680 MB | Time: 11.7s | GPU: 2.0% | VRAM: 781.0/16303.0 MB  
250/250 [=====] - 12s 47ms/step - loss: 0.4733 - accuracy: 0.7711 - val\_loss: 0.4756 - val\_accuracy: 0.7767  
Epoch 9/20  
249/250 [=====>.] - ETA: 0s - loss: 0.4618 - accuracy: 0.7810  
Epoch 9: saving model to checkpoints\_cnn\_2\weights\_epoch\_09.h5

```

| CPU: 1.9% | RAM: 15681 MB | Time: 11.4s | GPU: 2.0% | VRAM: 781.0/16303.0 MB
250/250 [=====] - 11s 46ms/step - loss: 0.4617 - accuracy: 0.7807 - val_loss: 0.5345 - val_accuracy: 0.7566
Epoch 10/20
249/250 [=====>.] - ETA: 0s - loss: 0.4509 - accuracy: 0.7844
Epoch 10: saving model to checkpoints_cnn_2\weights_epoch_10.h5
| CPU: 4.8% | RAM: 15701 MB | Time: 11.6s | GPU: 3.0% | VRAM: 781.0/16303.0 MB
250/250 [=====] - 12s 46ms/step - loss: 0.4512 - accuracy: 0.7841 - val_loss: 0.4737 - val_accuracy: 0.7802
Epoch 11/20
249/250 [=====>.] - ETA: 0s - loss: 0.4373 - accuracy: 0.8007
Epoch 11: saving model to checkpoints_cnn_2\weights_epoch_11.h5
| CPU: 2.2% | RAM: 15737 MB | Time: 11.5s | GPU: 4.0% | VRAM: 781.0/16303.0 MB
250/250 [=====] - 12s 46ms/step - loss: 0.4368 - accuracy: 0.8009 - val_loss: 0.4498 - val_accuracy: 0.7888
Epoch 12/20
249/250 [=====>.] - ETA: 0s - loss: 0.4283 - accuracy: 0.7984
Epoch 12: saving model to checkpoints_cnn_2\weights_epoch_12.h5
| CPU: 1.0% | RAM: 15726 MB | Time: 11.5s | GPU: 4.0% | VRAM: 781.0/16303.0 MB
250/250 [=====] - 12s 46ms/step - loss: 0.4284 - accuracy: 0.7983 - val_loss: 0.4527 - val_accuracy: 0.7918
Epoch 13/20
249/250 [=====>.] - ETA: 0s - loss: 0.4161 - accuracy: 0.8047
Epoch 13: saving model to checkpoints_cnn_2\weights_epoch_13.h5
| CPU: 2.8% | RAM: 15718 MB | Time: 11.5s | GPU: 5.0% | VRAM: 781.0/16303.0 MB
250/250 [=====] - 12s 46ms/step - loss: 0.4165 - accuracy: 0.8046 - val_loss: 0.4879 - val_accuracy: 0.7767
Epoch 14/20
249/250 [=====>.] - ETA: 0s - loss: 0.4046 - accuracy: 0.8146
Epoch 14: saving model to checkpoints_cnn_2\weights_epoch_14.h5
Restoring model weights from the end of the best epoch: 11.
| CPU: 2.1% | RAM: 15719 MB | Time: 11.6s | GPU: 2.0% | VRAM: 781.0/16303.0 MB
250/250 [=====] - 12s 46ms/step - loss: 0.4044 - accuracy: 0.8148 - val_loss: 0.4688 - val_accuracy: 0.7828
Epoch 14: early stopping

```

```

In [49]: model_3 = build_cnn(nb_conv=3)

checkpoint, early_stop, ckpt_dir = get_callbacks(3)

history_3 = model_3.fit(
    train_set,
    steps_per_epoch=train_set.samples // BATCH_SIZE,
    epochs=20,
    validation_data=test_set,
    validation_steps=test_set.samples // BATCH_SIZE,
    callbacks=[checkpoint, early_stop, PerformanceCallback()],
    verbose=1
)

```

Epoch 1/20  
249/250 [=====>.] - ETA: 0s - loss: 0.6688 - accuracy: 0.5833  
Epoch 1: saving model to checkpoints\_cnn\_3\weights\_epoch\_01.h5  
| CPU: 1.9% | RAM: 15724 MB | Time: 12.2s | GPU: 7.0% | VRAM: 781.0/16303.0 MB  
250/250 [=====] - 12s 48ms/step - loss: 0.6691 - accuracy: 0.5831 - val\_loss: 0.5921 - val\_accuracy: 0.6951  
Epoch 2/20  
249/250 [=====>.] - ETA: 0s - loss: 0.6036 - accuracy: 0.6679  
Epoch 2: saving model to checkpoints\_cnn\_3\weights\_epoch\_02.h5  
| CPU: 2.7% | RAM: 15732 MB | Time: 11.9s | GPU: 5.0% | VRAM: 781.0/16303.0 MB  
250/250 [=====] - 12s 48ms/step - loss: 0.6032 - accuracy: 0.6681 - val\_loss: 0.5703 - val\_accuracy: 0.7107  
Epoch 3/20  
249/250 [=====>.] - ETA: 0s - loss: 0.5748 - accuracy: 0.6963  
Epoch 3: saving model to checkpoints\_cnn\_3\weights\_epoch\_03.h5  
| CPU: 2.4% | RAM: 15729 MB | Time: 11.8s | GPU: 2.0% | VRAM: 833.0/16303.0 MB  
250/250 [=====] - 12s 47ms/step - loss: 0.5751 - accuracy: 0.6960 - val\_loss: 0.5256 - val\_accuracy: 0.7399  
Epoch 4/20  
249/250 [=====>.] - ETA: 0s - loss: 0.5270 - accuracy: 0.7324  
Epoch 4: saving model to checkpoints\_cnn\_3\weights\_epoch\_04.h5  
| CPU: 2.0% | RAM: 15726 MB | Time: 11.8s | GPU: 2.0% | VRAM: 781.0/16303.0 MB  
250/250 [=====] - 12s 47ms/step - loss: 0.5270 - accuracy: 0.7325 - val\_loss: 0.5045 - val\_accuracy: 0.7560  
Epoch 5/20  
249/250 [=====>.] - ETA: 0s - loss: 0.5022 - accuracy: 0.7551  
Epoch 5: saving model to checkpoints\_cnn\_3\weights\_epoch\_05.h5  
| CPU: 1.7% | RAM: 15699 MB | Time: 11.7s | GPU: 1.0% | VRAM: 781.0/16303.0 MB  
250/250 [=====] - 12s 47ms/step - loss: 0.5022 - accuracy: 0.7549 - val\_loss: 0.4900 - val\_accuracy: 0.7621  
Epoch 6/20  
249/250 [=====>.] - ETA: 0s - loss: 0.4769 - accuracy: 0.7668  
Epoch 6: saving model to checkpoints\_cnn\_3\weights\_epoch\_06.h5  
| CPU: 2.6% | RAM: 15702 MB | Time: 11.7s | GPU: 2.0% | VRAM: 781.0/16303.0 MB  
250/250 [=====] - 12s 47ms/step - loss: 0.4762 - accuracy: 0.7674 - val\_loss: 0.4569 - val\_accuracy: 0.7772  
Epoch 7/20  
249/250 [=====>.] - ETA: 0s - loss: 0.4580 - accuracy: 0.7816  
Epoch 7: saving model to checkpoints\_cnn\_3\weights\_epoch\_07.h5  
| CPU: 2.8% | RAM: 15725 MB | Time: 11.8s | GPU: 1.0% | VRAM: 781.0/16303.0 MB  
250/250 [=====] - 12s 47ms/step - loss: 0.4586 - accuracy: 0.7807 - val\_loss: 0.4387 - val\_accuracy: 0.7944  
Epoch 8/20  
249/250 [=====>.] - ETA: 0s - loss: 0.4457 - accuracy: 0.7868  
Epoch 8: saving model to checkpoints\_cnn\_3\weights\_epoch\_08.h5  
| CPU: 1.8% | RAM: 15696 MB | Time: 11.5s | GPU: 4.0% | VRAM: 833.0/16303.0 MB  
250/250 [=====] - 11s 46ms/step - loss: 0.4457 - accuracy: 0.7868 - val\_loss: 0.4399 - val\_accuracy: 0.7959  
Epoch 9/20  
249/250 [=====>.] - ETA: 0s - loss: 0.4183 - accuracy: 0.8021  
Epoch 9: saving model to checkpoints\_cnn\_3\weights\_epoch\_09.h5

| CPU: 1.9% | RAM: 15706 MB | Time: 11.5s | GPU: 1.0% | VRAM: 781.0/16303.0 MB  
250/250 [=====] - 12s 46ms/step - loss: 0.4184 - accuracy: 0.8019 - val\_loss: 0.4710 - val\_accuracy: 0.7737  
Epoch 10/20  
249/250 [=====>.] - ETA: 0s - loss: 0.4059 - accuracy: 0.8111  
Epoch 10: saving model to checkpoints\_cnn\_3\weights\_epoch\_10.h5  
| CPU: 1.7% | RAM: 15718 MB | Time: 11.5s | GPU: 2.0% | VRAM: 781.0/16303.0 MB  
250/250 [=====] - 12s 46ms/step - loss: 0.4056 - accuracy: 0.8111 - val\_loss: 0.4241 - val\_accuracy: 0.7989  
Epoch 11/20  
249/250 [=====>.] - ETA: 0s - loss: 0.3997 - accuracy: 0.8148  
Epoch 11: saving model to checkpoints\_cnn\_3\weights\_epoch\_11.h5  
| CPU: 2.9% | RAM: 15737 MB | Time: 11.6s | GPU: 3.0% | VRAM: 781.0/16303.0 MB  
250/250 [=====] - 12s 46ms/step - loss: 0.3994 - accuracy: 0.8149 - val\_loss: 0.4228 - val\_accuracy: 0.8029  
Epoch 12/20  
249/250 [=====>.] - ETA: 0s - loss: 0.3874 - accuracy: 0.8228  
Epoch 12: saving model to checkpoints\_cnn\_3\weights\_epoch\_12.h5  
| CPU: 1.2% | RAM: 15706 MB | Time: 11.5s | GPU: 5.0% | VRAM: 781.0/16303.0 MB  
250/250 [=====] - 11s 46ms/step - loss: 0.3878 - accuracy: 0.8225 - val\_loss: 0.4186 - val\_accuracy: 0.8024  
Epoch 13/20  
249/250 [=====>.] - ETA: 0s - loss: 0.3719 - accuracy: 0.8350  
Epoch 13: saving model to checkpoints\_cnn\_3\weights\_epoch\_13.h5  
| CPU: 2.0% | RAM: 15718 MB | Time: 11.5s | GPU: 1.0% | VRAM: 781.0/16303.0 MB  
250/250 [=====] - 11s 46ms/step - loss: 0.3716 - accuracy: 0.8351 - val\_loss: 0.4223 - val\_accuracy: 0.8130  
Epoch 14/20  
249/250 [=====>.] - ETA: 0s - loss: 0.3586 - accuracy: 0.8414  
Epoch 14: saving model to checkpoints\_cnn\_3\weights\_epoch\_14.h5  
| CPU: 2.2% | RAM: 15705 MB | Time: 11.5s | GPU: 5.0% | VRAM: 781.0/16303.0 MB  
250/250 [=====] - 12s 46ms/step - loss: 0.3585 - accuracy: 0.8413 - val\_loss: 0.4069 - val\_accuracy: 0.8085  
Epoch 15/20  
250/250 [=====] - ETA: 0s - loss: 0.3501 - accuracy: 0.8439  
Epoch 15: saving model to checkpoints\_cnn\_3\weights\_epoch\_15.h5  
| CPU: 1.2% | RAM: 15750 MB | Time: 11.7s | GPU: 5.0% | VRAM: 781.0/16303.0 MB  
250/250 [=====] - 12s 47ms/step - loss: 0.3501 - accuracy: 0.8439 - val\_loss: 0.3980 - val\_accuracy: 0.8216  
Epoch 16/20  
249/250 [=====>.] - ETA: 0s - loss: 0.3381 - accuracy: 0.8478  
Epoch 16: saving model to checkpoints\_cnn\_3\weights\_epoch\_16.h5  
| CPU: 1.6% | RAM: 15761 MB | Time: 11.5s | GPU: 3.0% | VRAM: 781.0/16303.0 MB  
250/250 [=====] - 12s 46ms/step - loss: 0.3380 - accuracy: 0.8478 - val\_loss: 0.4236 - val\_accuracy: 0.8100  
Epoch 17/20  
249/250 [=====>.] - ETA: 0s - loss: 0.3332 - accuracy: 0.8564  
Epoch 17: saving model to checkpoints\_cnn\_3\weights\_epoch\_17.h5  
| CPU: 1.8% | RAM: 15733 MB | Time: 11.5s | GPU: 1.0% | VRAM: 781.0/16303.0 MB  
250/250 [=====] - 11s 46ms/step - loss: 0.3328 - accuracy: 0.8568 - val\_loss: 0.4226 - val\_accuracy: 0.8155  
Epoch 18/20

```

249/250 [=====>.] - ETA: 0s - loss: 0.3142 - accuracy: 0.8579
Epoch 18: saving model to checkpoints_cnn_3\weights_epoch_18.h5
| CPU: 2.1% | RAM: 15730 MB | Time: 11.6s | GPU: 1.0% | VRAM: 781.0/16303.0 MB
250/250 [=====] - 12s 47ms/step - loss: 0.3138 - accuracy: 0.8583 - val_loss: 0.3847 - val_accuracy: 0.8367
Epoch 19/20
249/250 [=====>.] - ETA: 0s - loss: 0.3071 - accuracy: 0.8630
Epoch 19: saving model to checkpoints_cnn_3\weights_epoch_19.h5
| CPU: 1.1% | RAM: 15722 MB | Time: 11.6s | GPU: 3.0% | VRAM: 781.0/16303.0 MB
250/250 [=====] - 12s 46ms/step - loss: 0.3074 - accuracy: 0.8629 - val_loss: 0.3756 - val_accuracy: 0.8392
Epoch 20/20
249/250 [=====>.] - ETA: 0s - loss: 0.2918 - accuracy: 0.8732
Epoch 20: saving model to checkpoints_cnn_3\weights_epoch_20.h5
| CPU: 2.4% | RAM: 15619 MB | Time: 11.6s | GPU: 3.0% | VRAM: 781.0/16303.0 MB
250/250 [=====] - 12s 46ms/step - loss: 0.2924 - accuracy: 0.8733 - val_loss: 0.5256 - val_accuracy: 0.7802

```

```

In [50]: import matplotlib.pyplot as plt

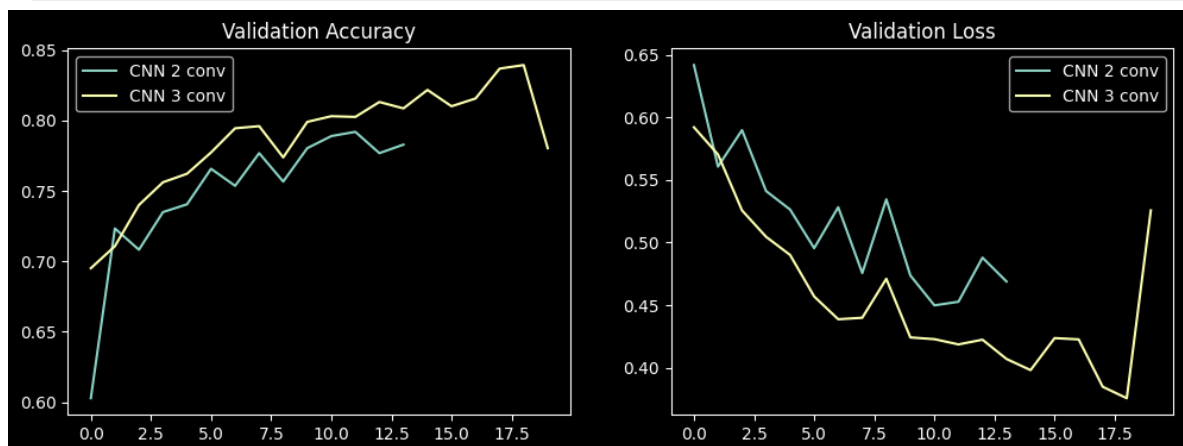
plt.figure(figsize=(12,4))

plt.subplot(1,2,1)
plt.plot(history_2.history["val_accuracy"], label="CNN 2 conv")
plt.plot(history_3.history["val_accuracy"], label="CNN 3 conv")
plt.title("Validation Accuracy")
plt.legend()

plt.subplot(1,2,2)
plt.plot(history_2.history["val_loss"], label="CNN 2 conv")
plt.plot(history_3.history["val_loss"], label="CNN 3 conv")
plt.title("Validation Loss")
plt.legend()

plt.show()

```



## Comparaison CNN 2 convolutions vs CNN 3 convolutions (Dogs & Cats)

L'entraînement des deux architectures montre que l'ajout d'une troisième couche de convolution améliore les performances du modèle, tout en conservant un entraînement



stable et bien régularisé grâce à l'early stopping.

CNN 2 convolutions :

- Précision maximale sur l'ensemble de validation : ~79,2 %
- Perte de validation minimale : ~0,45
- Early stopping déclenché à l'epoch 14, meilleur modèle restauré à l'epoch 12

CNN 3 convolutions :

- Précision maximale sur l'ensemble de validation : ~83,9 %
- Perte de validation minimale : ~0,38
- Entraînement mené jusqu'à l'epoch 20, avec performances optimales observées entre les epochs 18 et 19

Le graphe de précision de validation montre que le CNN à 3 convolutions converge plus régulièrement et atteint une précision systématiquement supérieure à celle du CNN à 2 convolutions sur la majorité des epochs. De plus, le graphe de perte de validation indique une diminution plus marquée et plus stable pour le modèle à 3 convolutions, traduisant une meilleure capacité de généralisation.

Lors de l'entraînement des modèles CNN à 2 et 3 convolutions, l'utilisation des ressources matérielles reste faible et stable. Le CPU est peu sollicité (environ 1 à 5 %), le GPU présente une utilisation modérée avec quelques pics ponctuels (jusqu'à ~30–40 % en début d'entraînement pour le CNN à 2 convolutions), la RAM reste stable autour de 15 à 16 Go et la VRAM demeure inférieure à 1 Go. Ces observations montrent que l'entraînement n'est pas limité par le matériel et que les performances obtenues sont liées aux choix d'architecture.