

# **Laboratoire no 4 ARN**

**Deep Neural Network**

**9 Mai 2023**

**Étudiants :**

Jérémie Santoro & Théo Coudaudier

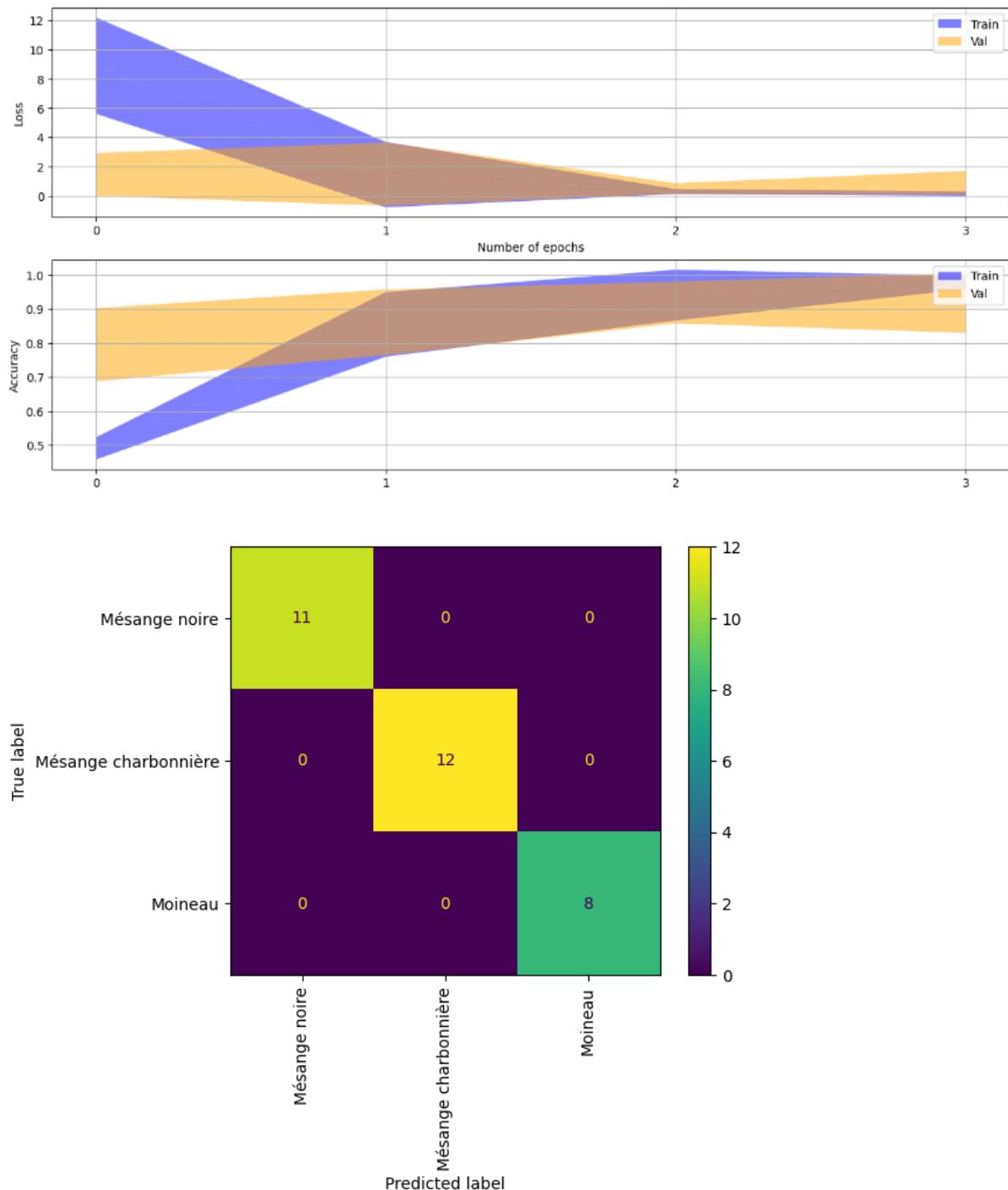
# Table des matières

<b>1. Part 1</b>	<b>3</b>
<b>1. Introduction</b>	<b>4</b>
<b>2. The problem</b>	<b>5</b>
<b>3. Data preparation</b>	<b>7</b>
<b>4. Model creation</b>	<b>8</b>
<b>5. Results</b>	<b>9</b>
<b>6. Conclusions</b>	<b>17</b>

# 1. Part 1

Il m'a suffit d'ajouter une couche dense avec 20 neurones et ReLu pour obtenir des résultats "parfaits" comme on peut voir avec la matrice de confusion notamment.

J'ai également freeze les base model layers.



# 1. Introduction

## **describe the context of your application and its potential uses**

L'application va servir à catégoriser 6 de mes doudous (classification d'images):

1. Chat habillé (maigre)
2. Chat tout nu (gros)
3. Requin marteau
4. Requin gris
5. Grenouille
6. Clone (Star Wars)
7. Souris
8. Rat foncé
9. Rat clair

L'application n'a pas réellement d'utilité pratique, mais est assez ludique 😊

Pour ce faire nous allons utiliser Keras TensorFlow pour notamment faire nos modèles.

## **briefly describe how you are going to proceed (methodology), that is, what data are you going to collect (your own pictures of.... , maybe extra-pictures collected from the web, etc.)**

Pour la méthodologie je vais tout simplement suivre les TODO du part2.ipynb, c'est à dire:

- Importer mes photos
- Data augmentation
- Designer le model sur la base de MobileNetV2 (Freeze certains layers pour garder les propriétés du modèle pré-entraîné, ...)
- Cross validation (model training)
- Utiliser grad-cam pour voir comment notre modèle classifie
- Comprendre avec grad-cam pourquoi certaines classes fonctionnent moins bien que d'autres

## **what methods are you going to use (e.g., CNNs, transfer learning) .**

- Utilisation du transfer learning depuis MobileNetV2 (en freezant des layers)
- Utilisation de CNN (pratique pour la reconnaissance d'images)

## 2. The problem

describe what are the classes you are learning to detect using a CNN

Nom	Image
<b>Chat habillé</b> ( <i>pismice_maigre</i> )	
<b>Chat tout nu</b> ( <i>pismice_gros</i> )	
<b>Requin marteau</b> ( <i>requin_marteau</i> )	
<b>Requin gris</b> ( <i>requin_gris</i> )	
<b>Grenouille</b> ( <i>crapidouille</i> )	

**Clone** (*clone*)



**Souris** (*souris*)



**Rat foncé** (*rat\_fonce*)



**Rat clair** (*rat\_clair*)

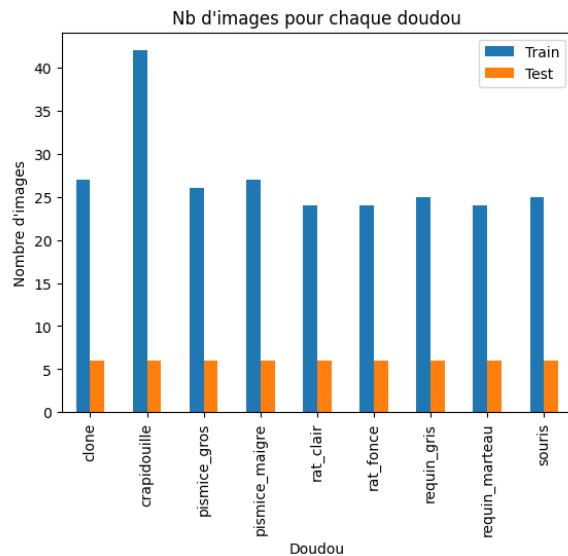


**describe the database you collected, show the number of images per class or a histogram of classes (is it a balanced or an unbalanced dataset? Small or big Data ?)**

J'ai ~30 images par classe (~290 en tout), ce qui fait que le dataset est balanced (car nombre égal d'échantillons pour chaque classe) et qu'il s'agit de small data.

J'ai pris les photos sur mon lit pour tenter d'avoir un fond qui n'influence pas la classification.

Remarque: j'ai + de grenouilles (crapidouille), car je n'ai pas réussi à compter jusqu'à 30 en prenant mes photos.



**and provide some examples showing the intra-class diversity and the apparent difficulty for providing a solution (inter-class similarity).**

Il s'agit de doudous qui font tous un peu près la même taille et qui ont la même matière (petits poils), certains se ressemblent plus que d'autres comme par exemple :

- Les 2 requins (car il s'agit du même animal)
- Les 2 chats (littéralement le même doudou mais un est habillé)
- Les 2 rats (seul la couleur change)
- La couleur de la souris, du requin gris et du rat clair sont la même (gris clair)

### 3. Data preparation

**describe the pre-processing steps you needed (e.g., resizing, normalization, filtering of collected images, perhaps you discarded some data).**

Toutes les images sont mises sous (224x224) en cropant vers l'aspect ratio.

J'ai ensuite effectué une image augmentation en appliquant divers filtres tels que du zoom, des rotations, du flip et du contraste.

J'ai hésité à enlever les images floues de mon dataset mais pour finir je ne l'ai pas fait car je me suis dit que ce serait "un peu comme" une data augmentation.

**Describe the train, validation and test datasets split.**

Sur mes à peu près 30 photos par classe, j'en ai gardé à chaque fois 6 pour le set de test. Puis pour le reste j'ai gardé 80% pour l'entraînement et 20% pour la validation

## 4. Model creation

describe how did you proceed to come up with a final model (model selection methodology, hyper-parameter exploration, cross-validation)

a. What hyperparameters did you choose (nb epochs, optimizer, learning rate, ...) ?

Nb d'epochs = 4

Optimizer = RMSProp

Learning rate = Default de RMSProp

Fonction d'activation = ReLu

b. What is the architecture of your final model ? How many trainable parameters does it have?

1 Dense layer = 1 layer de 50 neurones

Trainable params = 64509

```
Model: "model"
=====
Layer (type)      Output Shape       Param #  Connected to
=====
input_1 (InputLayer) [(None, 224, 224, 3)]   0        []
Conv1 (Conv2D)     (None, 112, 112, 32)    864      ['input_1[0][0]']
bn_Conv1 (BatchNormalizati on) (None, 112, 112, 32) 128      ['Conv1[0][0]']
Conv1_relu (ReLU)  (None, 112, 112, 32)    0        ['bn_Conv1[0][0]']
expanded_conv_depthwise (D epthwiseConv2D) (None, 112, 112, 32) 288      ['Conv1_relu[0][0]']
expanded_conv_depthwise_BN (BatchNormalization) (None, 112, 112, 32) 128      ['expanded_conv_depthwise[0][0]']
expanded_conv_depthwise_re lu (ReLU)          (None, 112, 112, 32) 0        ['expanded_conv_depthwise_BN[0][0]']
expanded_conv_project (Con v2D)                (None, 112, 112, 16) 512      ['expanded_conv_depthwise_relu[0][0]']

...
Total params: 2322493 (8.86 MB)
Trainable params: 64509 (251.99 KB)
Non-trainable params: 2257984 (8.61 MB)
```

c. How did you perform the transfer learning ?

Il suffit de freeze les layers (de préférence les premières et pas les dernières

```
# TODO: freeze layers
nb_layers = len(base_model.layers)
for layer in model.layers[:nb_layers - 1]:
    layer.trainable = False
```

Explain why did you use transfer learning, e.g., what is the advantage of using transfer learning for this problem and why it might help ?

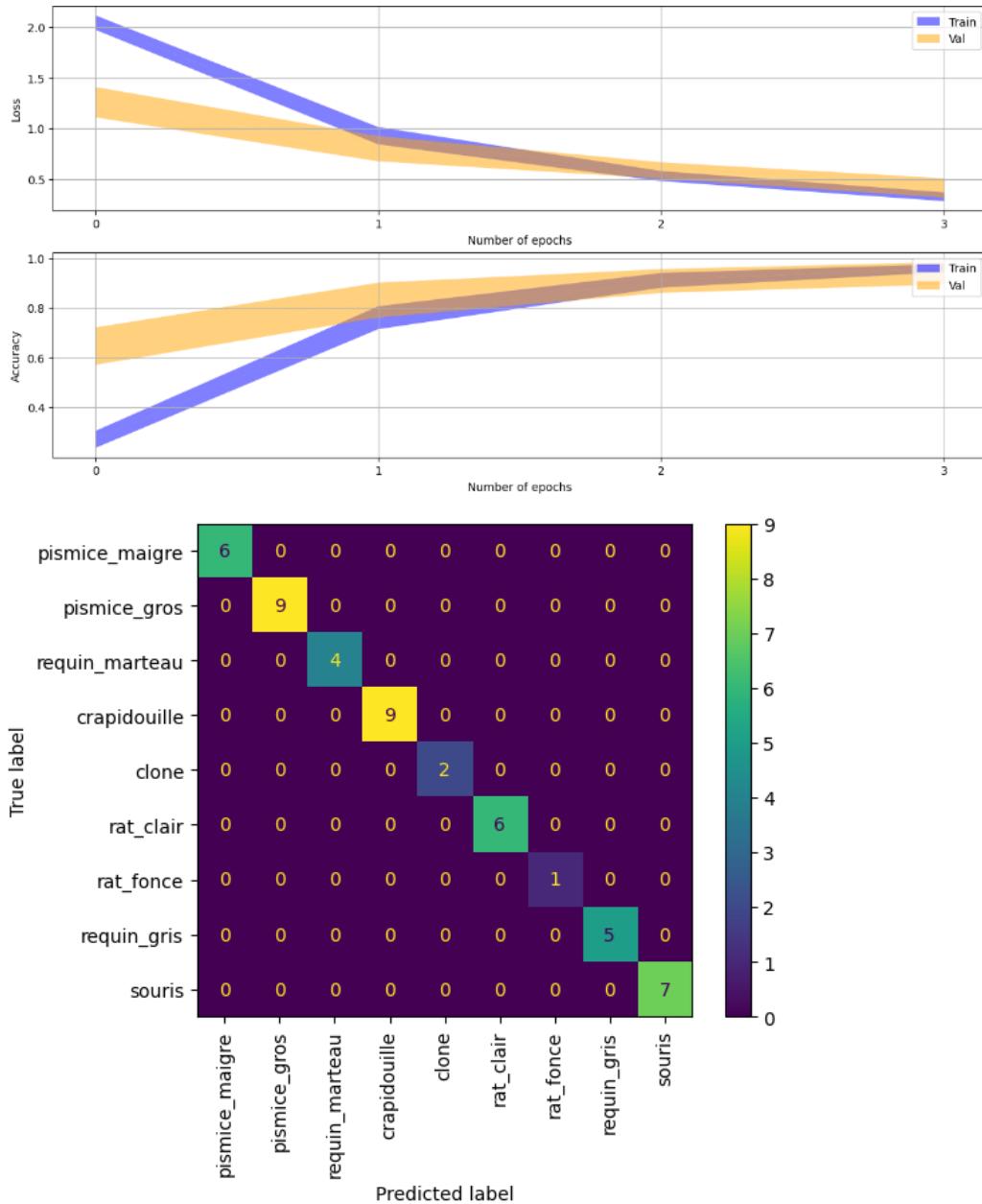
C'est utile si on a un petit dataset notamment car on possède déjà un modèle qui correspond plus ou moins à nos besoins. Avec un petit dataset comme ça il aurait été beaucoup plus fastidieux d'ajuster correctement les poids.

On aura donc un temps d'apprentissage moins long.

## 5. Results

describe the experiments you performed with the model both off-line (running in your notebooks with your own-collected images) and on-line (running in the smartphone and processing images captured in the “wild”).

### a. Provide your plots and confusion matrices



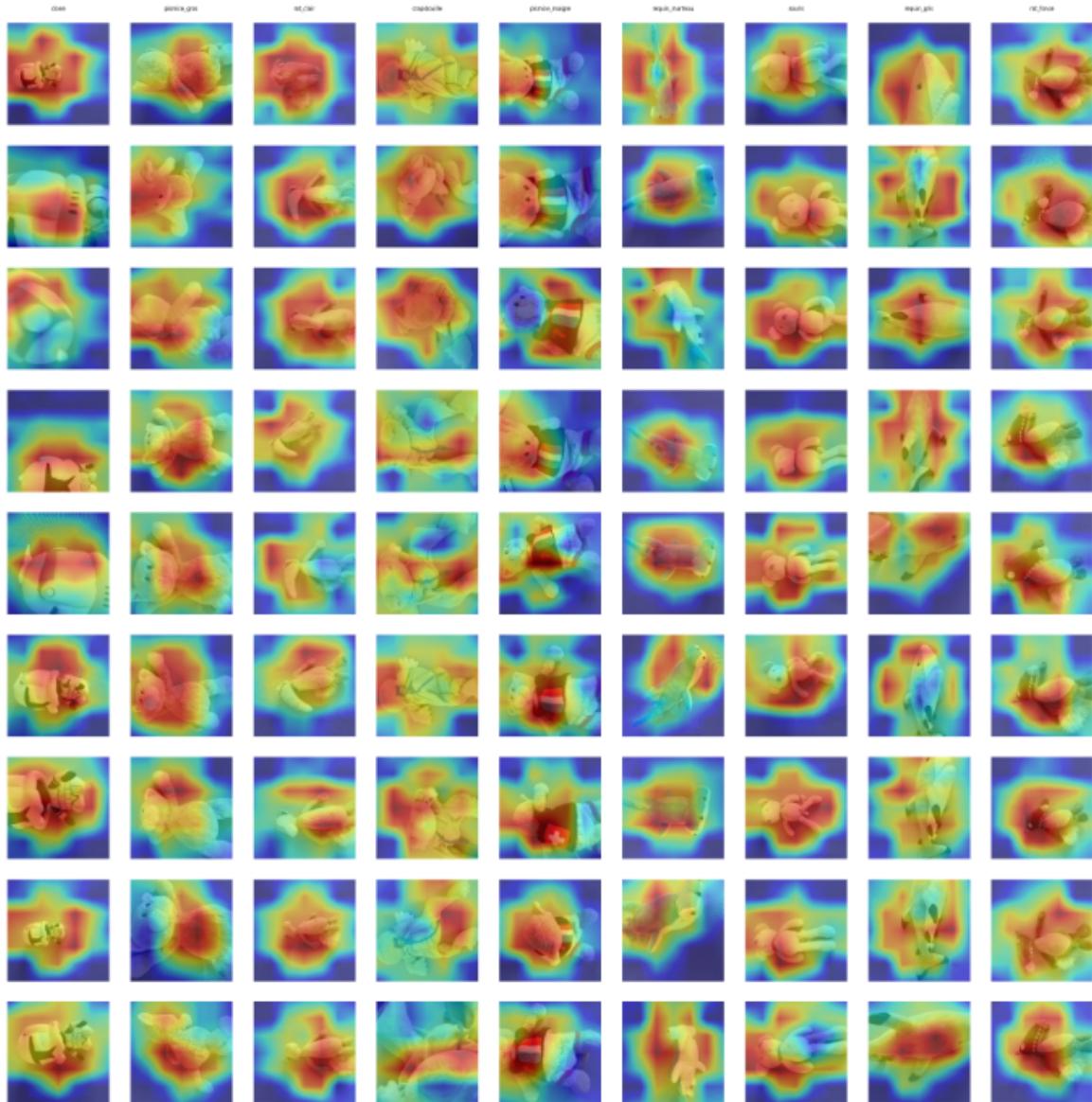
### b. Provide the f-score you obtain for each of your classes.

```
F-score pismice_maire: 1.0
F-score pismice_gros: 1.0
F-score requin_marteau: 1.0
F-score rapidouille: 1.0
F-score clone: 1.0
F-score rat_clair: 1.0
F-score rat_fonce: 1.0
F-score requin_gris: 1.0
F-score souris: 1.0
F1 score global: 1.0
```

**c. Provide the results you have after evaluating your model on the test set. Comment if the performance on the test set is close to the validation performance. What about the performance of the system in the real world ?**

Je pense que mon système classifie très bien mes doudous, mais étrangement quand le "real world" les performances ne sont clairement pas à la hauteur de ce qu'on penserait vu les plots au dessus.

**d. Present an analysis of the relevance of the trained system using the Class Activation Map methods (grad-cam)**



En voyant ces images j'ai vraiment l'impression que ça va marcher (bien que j'ai un peu peur quand je vois que la heat map fait des fois presque toute l'image) car je vois qu'il analyse bien les motifs spécifiques à chaque doudou et ne prend pas mon drap par exemple.

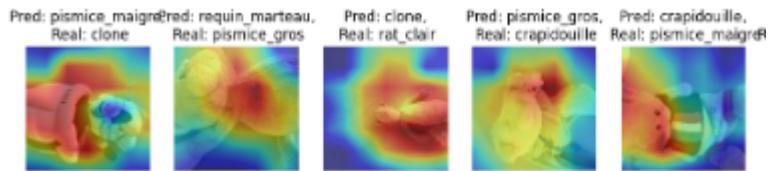
Quelques exemples que je trouve intéressants:

- Pour les chats habillés (colonne 5), il a l'air de mettre une intensité sur les drapeaux de son t-shirt, bien que sur la 3ème il omelette complètement la tête du doudou.
- Pour les requins (colonnes 6 et 8), il reconnaît les nageoires

e. Provide some of your misclassified images (test set and real-world tests) and comment on those errors.

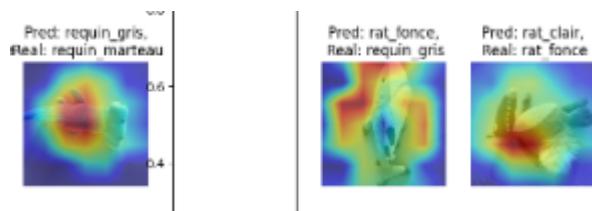
**Test set:**

Je possède beaucoup qui ne font pas de sens du tout à mes yeux, notamment la 2ème image où il reconnaît la grenouille en se basant sur la poitrine du chat sans habits.



Parfois je peux par contre comprendre pourquoi:

- Il confond les 2 requins à cause des nageoires
- Un requin avec le rat gris car ils ont la même couleur.
- Le rat clair avec le rat foncé car ils ont la même forme (à l'exception des dents)



## Real-world set:

 <table border="1"><thead><tr><th>Category</th><th>Confidence (%)</th></tr></thead><tbody><tr><td>rat_clair</td><td>73,0%</td></tr><tr><td>clone</td><td>20,7%</td></tr><tr><td>crapidouille</td><td>2,4%</td></tr></tbody></table>	Category	Confidence (%)	rat_clair	73,0%	clone	20,7%	crapidouille	2,4%	Correct
Category	Confidence (%)								
rat_clair	73,0%								
clone	20,7%								
crapidouille	2,4%								
 <table border="1"><thead><tr><th>Category</th><th>Confidence (%)</th></tr></thead><tbody><tr><td>rat_clair</td><td>63,6%</td></tr><tr><td>clone</td><td>11,1%</td></tr><tr><td>rat_fonce</td><td>6,4%</td></tr></tbody></table>	Category	Confidence (%)	rat_clair	63,6%	clone	11,1%	rat_fonce	6,4%	Confond avec le rat clair au dessus
Category	Confidence (%)								
rat_clair	63,6%								
clone	11,1%								
rat_fonce	6,4%								
 <table border="1"><thead><tr><th>Category</th><th>Confidence (%)</th></tr></thead><tbody><tr><td>souris</td><td>26,7%</td></tr><tr><td>rat_fonce</td><td>20,2%</td></tr><tr><td>pismice_maigre</td><td>18,8%</td></tr></tbody></table>	Category	Confidence (%)	souris	26,7%	rat_fonce	20,2%	pismice_maigre	18,8%	Ok, mais pas beaucoup de confiance
Category	Confidence (%)								
souris	26,7%								
rat_fonce	20,2%								
pismice_maigre	18,8%								



Seulement 13%, je n'arrive pas à comprendre pourquoi il reconnaît un requin marteau, peut être les membres qui sortent à la manière de nageoires ?

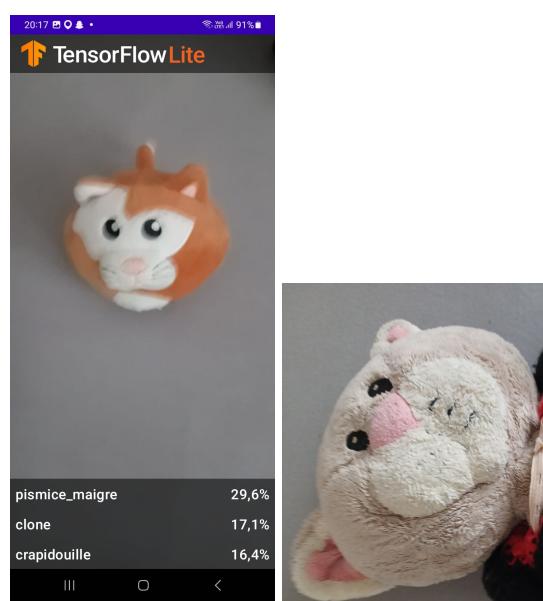


Vue de dessus c'est légèrement mieux

<table border="1"> <thead> <tr> <th></th> <th></th> </tr> </thead> <tbody> <tr> <td>pismice_maigre</td> <td>93,2%</td> </tr> <tr> <td>rat_fonce</td> <td>2,8%</td> </tr> <tr> <td>requin_gris</td> <td>1,6%</td> </tr> </tbody> </table> <table border="1"> <thead> <tr> <th></th> <th></th> </tr> </thead> <tbody> <tr> <td>pismice_gros</td> <td>85,8%</td> </tr> <tr> <td>requin_gris</td> <td>6,6%</td> </tr> <tr> <td>requin_marteau</td> <td>2,3%</td> </tr> </tbody> </table>			pismice_maigre	93,2%	rat_fonce	2,8%	requin_gris	1,6%			pismice_gros	85,8%	requin_gris	6,6%	requin_marteau	2,3%	<p>Aucun sens de confondre avec autant de confiance avec le chat habillé. Puis de confondre la grenouille avec le chat tout nu</p>
pismice_maigre	93,2%																
rat_fonce	2,8%																
requin_gris	1,6%																
pismice_gros	85,8%																
requin_gris	6,6%																
requin_marteau	2,3%																
<table border="1"> <thead> <tr> <th></th> <th></th> </tr> </thead> <tbody> <tr> <td>rat_fonce</td> <td>93,4%</td> </tr> <tr> <td>requin_gris</td> <td>2,3%</td> </tr> <tr> <td>pismice_maigre</td> <td>2,2%</td> </tr> </tbody> </table>			rat_fonce	93,4%	requin_gris	2,3%	pismice_maigre	2,2%	<p>Erreur sûrement à cause des couleurs grises et noires et nageoire peut-être pas assez mises en évidence</p>								
rat_fonce	93,4%																
requin_gris	2,3%																
pismice_maigre	2,2%																



Erreur car confond avec le requin du dessus



J'ai essayé de tester un doudou qui ne fait pas parti de mon dataset et je vois qu'il confond avec mon chat surement à cause du nez rose, moustaches, yeux et oreilles très similaire



**f. Based on your results how could you improve your dataset ?**

- Etant donné que j'ai eu de mauvais résultats au "live test", je pense que la diversité de mes photos dans mon dataset n'était pas assez variée pour correspondre à plusieurs situations (lumière du jour, fond, position, ...)
- Avoir choisi 9 classes était peut-être trop ambitieux.

**g. Observe which classes are confused. Does it surprise you? In your opinion, what can cause those confusions ?**

En dehors des exemples présentés plus tôt, j'ai remarqué que le "live test" avait tendance à confondre les doudous qui possèdent des dents pointus (les 2 requins et le rat foncé), des nageoires, la même couleur ou la même forme dans les cas des 2 rats clair et foncé.

**What happens when you use your embedded system to recognize objects that don't belong to any classes in your dataset ? How does your system work if your object is placed in a different background ?**

J'ai testé plusieurs autres doudous, dont celui présenté dans l'exemple au-dessus. Le système a tendance à catégoriser les nouveaux venus en 2 "super-classes" différents: ceux possédant des nageoires et ceux possédant des jambes et bras.

## 6. Conclusions

**finalize your report with some conclusions, summarize your results, mention the limits of your system and potential future work.**

Je suis un peu déçu par le live test car je m'attendais à beaucoup mieux au vu des résultats lorsque j'ai testé avec mon test set (100% de réussite).

Je me retrouve avec une classification très mauvaise avec mon téléphone, sûrement car les photos que j'ai prises pour mon train, validation et test set ont été prises au même moment de la journée en l'espace de 10 minutes, les photos se ressemblent donc beaucoup et il est possible que le système soit perdu si je ne suis pas dans les exactes paramètres dans lesquelles je me trouvais au moment de la prise des photos.

L'hypothèse selon moi la plus probable est que j'ai fait de l'overfitting sur les données de base.

J'ai hâte de recevoir le feedback du labo pour savoir ce que je peux améliorer 😊

Remarque: J'ai également testé avec d'autres téléphones mais ça n'a rien changé aux mauvais résultats.

Merci aux acteurs:

