

ENTRAÎNER UN MODELE SUR SES PROPRIETES DONNEES

Sous titre

Prénom Nom - Date



UNIVERSITÉ DE
MONTPELLIER

Inserm



Contrats de plan
ÉTAT-RÉGION

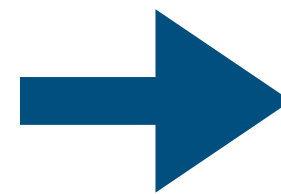


anr[®]
agence nationale
de la recherche

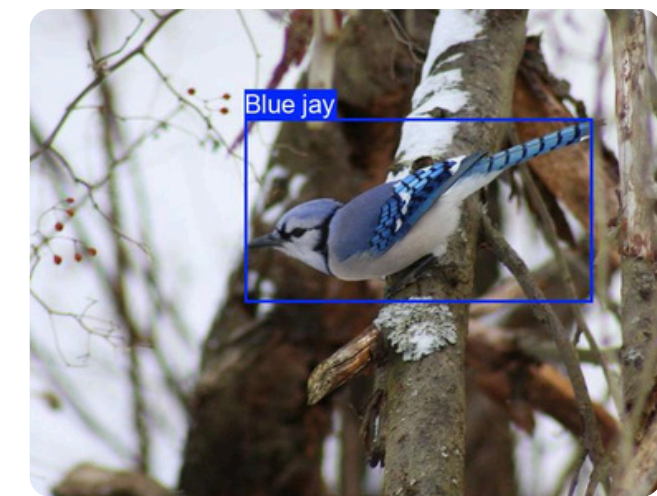
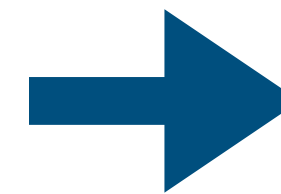


INTRODUCTION

Dans ce tutoriel on va voir comment faire de la détection et reconnaissance d'objets sur une image. On va donc entraîner un modèle d'intelligence artificielle pour qu'il soit capable de reconnaître ce que l'on souhaite et l'encadrer dans une boîte (appelée "boîte englobante"). Concrètement, une fois entraîné, on pourra faire ça :



Modèle d'intelligence artificielle
(Yolov26)



Donc on aura un outil non seulement capable de trouver spatialement quelque chose mais aussi de le reconnaître.

SOMMAIRE

1 Annotation
des données

2 Entraînement
du modèle

3 Résultats et
évaluation du
modèle

4 Amélioration
du modèle

5 Titre de
la partie

6 Titre de
la partie

INTRODUCTION

Notion de classe

Entraîner un modèle consiste à lui donner beaucoup d'images où l'objet d'intérêt est déjà encadré pour qu'il puisse comprendre par lui-même ce qui définit notre objet (un machine et un humain ne voient pas le monde pareil).

Les objets d'intérêt doivent porter un nom qui désigne leur catégorie d'appartenance que l'on appelle **classe**. Donc par exemple on peut avoir une classe "chat" et une classe "chien", mais on peut aussi avoir une classe "labrador" et "chihuahua". Vous pouvez choisir absolument ce que vous voulez, que ce soit très général ou très spécifique.

Nous utiliserons dorénavant beaucoup le mot classe, donc il faut bien se rappeler que ce mot correspond simplement aux catégories d'objets que l'on souhaite détecter.

INTRODUCTION

Diversité des données

Il est important de comprendre que le modèle, une fois entraîné sur des données, ne sera efficace que sur des données similaires aux données d'entraînement.

Par exemple : si vous n'aviez qu'une classe "chien" et que vos données ne montrent que des chiens dans la neige, si vous donnez une image de chien dans une prairie, il ne le reconnaîtra pas forcément voire pas du tout.

À retenir : ayez une base de données à l'image de ce que vous aurez besoin de faire détecter.

Concernant le nombre d'images que le modèle a besoin il n'y a pas de nombre exacte, ainsi vous pouvez lire cette documentation qui est dédiée au sujet :

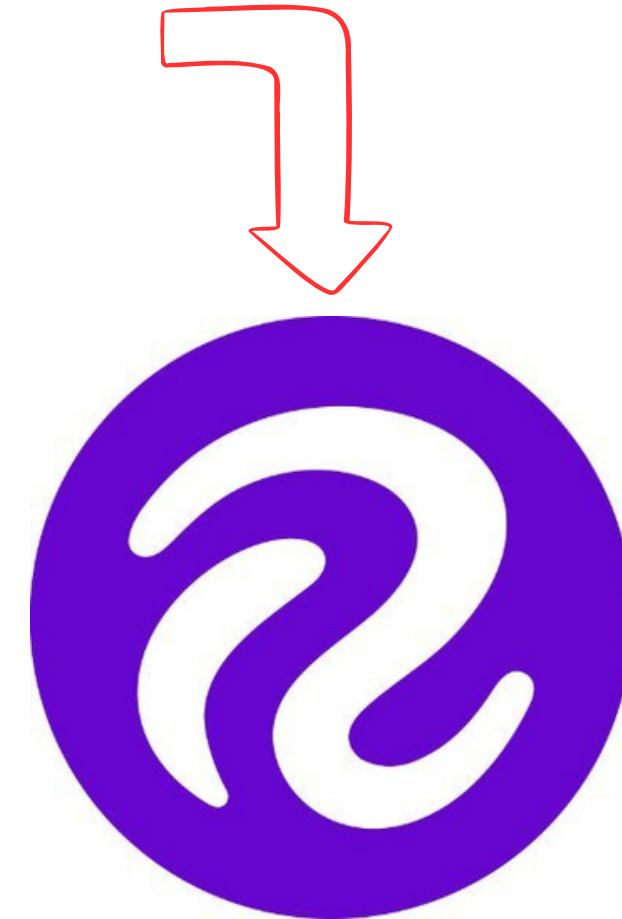
<https://blog.roboflow.com/images-train-model/>

Si vous voulez démarrer vite, commencez avec 100 images, c'est un bon début.

ANNOTATIONS DES DONNÉES

DIFFÉRENTS OUTILS DISPONIBLES

Il existe plusieurs outils d'annotations gratuits comme CVAT, LabelStudio ou Roboflow et ici nous utiliserons et détaillerons l'utilisation de ce dernier.



Nous avancerons pas à pas à travers Roboflow, donc vous êtes invités à vous rendre sur le site "Roboflow.com" puis à suivre les instructions des pages suivantes.

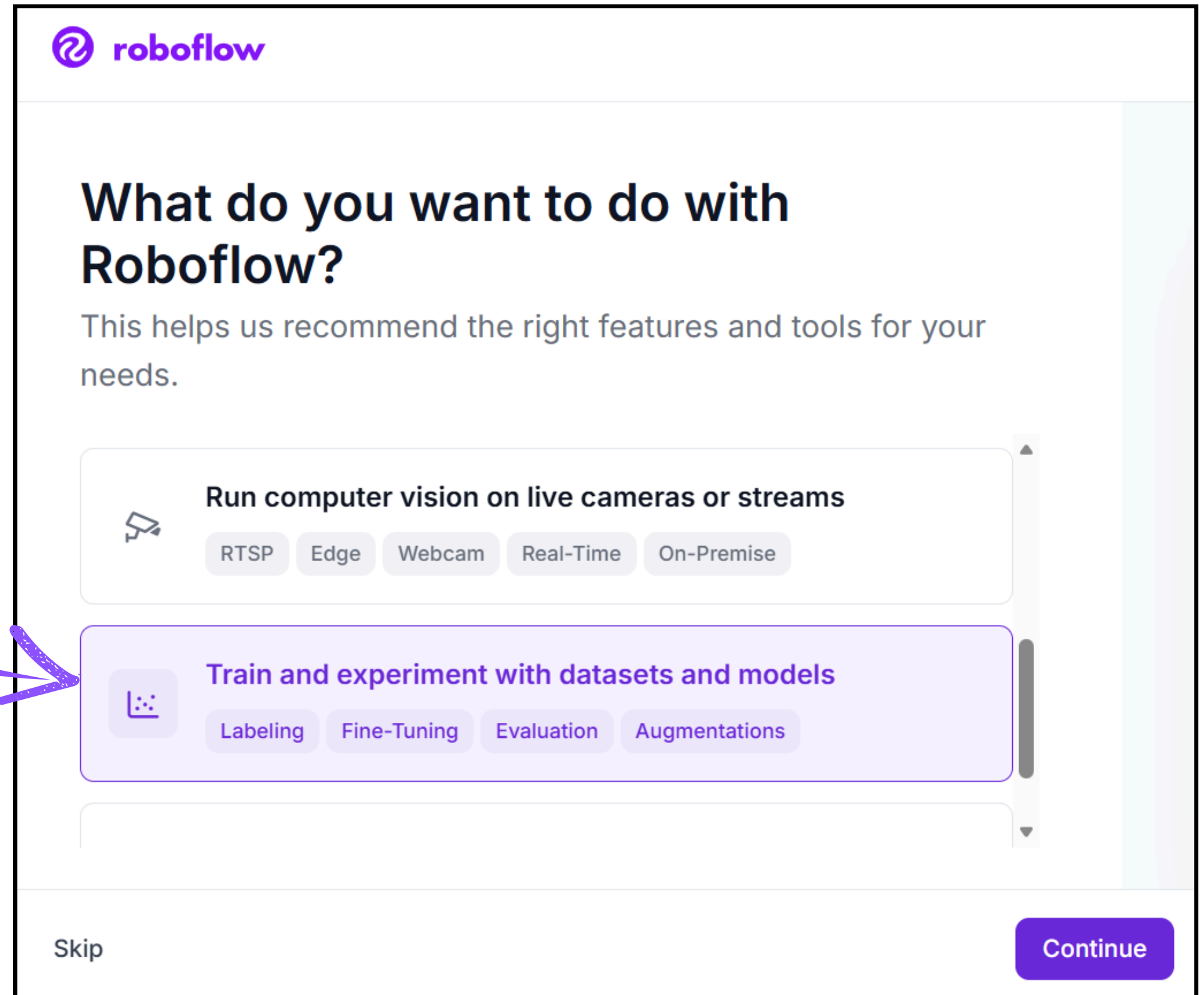
PAS À PAS

Commencez par vous faire un compte, c'est gratuit, puis nommez votre espace de travail

Lorsque cette fenêtre apparaît, sélectionnez cette rubrique :

Invitez des collègues dans votre espace à votre convenance.

Roboflow devrait vous proposer 14 jours d'essai de l'abonnement payant gratuitement.



roboflow

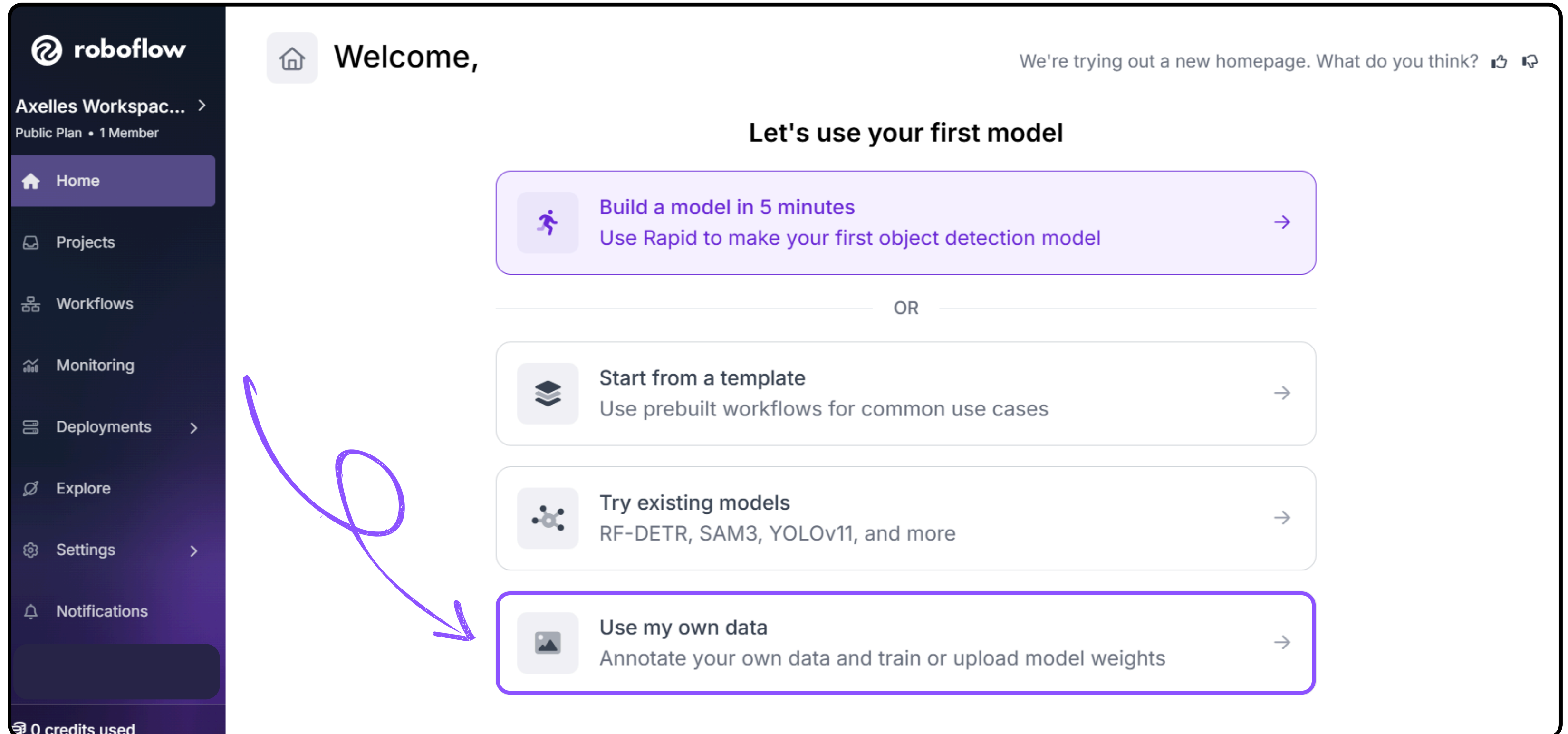
What do you want to do with Roboflow?

This helps us recommend the right features and tools for your needs.

- Run computer vision on live cameras or streams
 - RTSP
 - Edge
 - Webcam
 - Real-Time
 - On-Premise
- Train and experiment with datasets and models**
 - Labeling
 - Fine-Tuning
 - Evaluation
 - Augmentations

Skip Continue

PAS À PAS



roboflow

Axelles Workspac... >
Public Plan • 1 Member

- Home
- Projects
- Workflows
- Monitoring
- Deployments >
- Explore
- Settings >
- Notifications

0 credits used

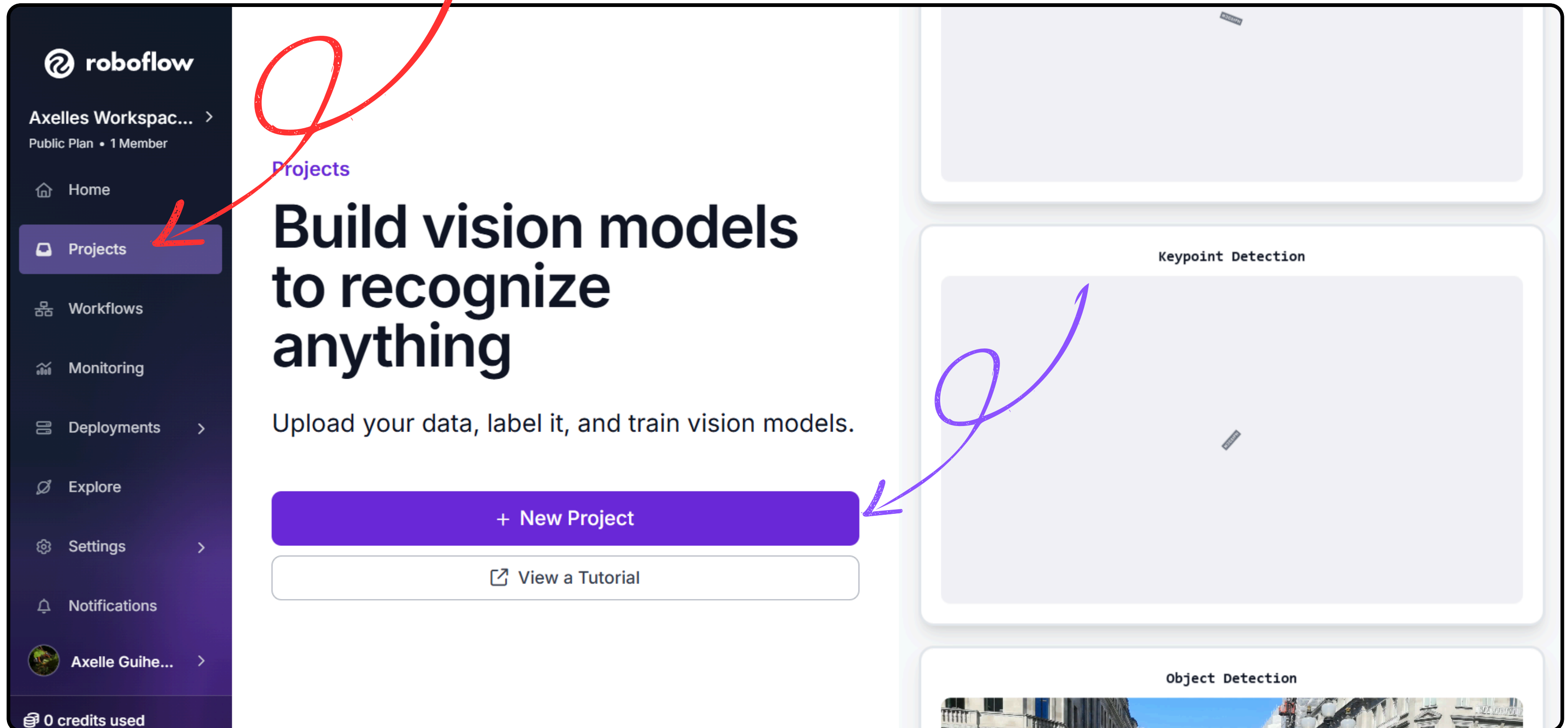
Welcome, We're trying out a new homepage. What do you think? 👍 🔄

Let's use your first model

- Build a model in 5 minutes**
Use Rapid to make your first object detection model →
- OR
- Start from a template
Use prebuilt workflows for common use cases →
- Try existing models
RF-DETR, SAM3, YOLOv11, and more →
- Use my own data**
Annotate your own data and train or upload model weights →

Si votre page d'accueil ne ressemble pas à ça, allez à la diapo suivante.

PAS À PAS



roboflow

Axelles Workspac... >
Public Plan • 1 Member

- Home
- Projects**
- Workflows
- Monitoring
- Deployments >
- Explore
- Settings >
- Notifications
- Axelle Guihe... >

0 credits used

Projects

Build vision models to recognize anything

Upload your data, label it, and train vision models.

[+ New Project](#)

[View a Tutorial](#)

Keypoint Detection


Object Detection

PAS À PAS


Get Started

- Use Prebuilt Templates
- Try Existing Models
- Use My Own Data**

We Recommend




Roboflow Rapid
Build object detection models in minutes



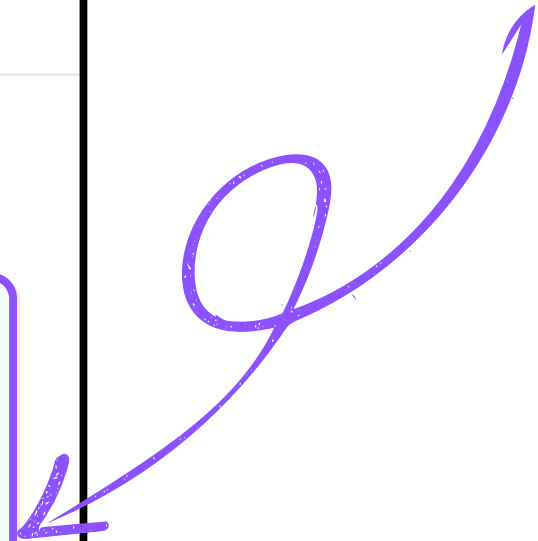
Build a Custom Model
Upload, label, and train with Roboflow

Other Options



Upload a Model
Use custom-trained model weights

Want help building?
[Talk to someone at Roboflow](#)



PAS À PAS

Durant l'essai gratuit, vous pouvez choisir "Private", sinon ce sera "Public"

Let's create your project.

Workspace > Private Trial My First Project

1 Project Name

My First Project

2 Annotation Group ?

objects

Visibility ?

Private

Public

3 Project Type

Object Detection

Identify objects and their positions with bounding boxes.

Bounding Boxes

Counts

Tracking

4

Continue with Private

PAS À PAS

Use Traditional Model Builder Instead



Le projet va se construire. Il vous faut ensuite fournir vos images.

Drag and drop file(s) to upload, or:

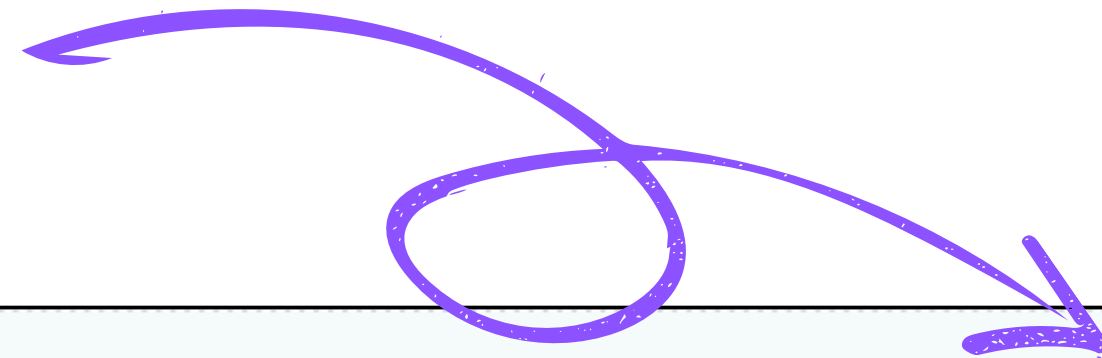
 Select File(s)

 Select Folder



PAS À PAS

Quand vous avez bien mis toutes vos images, validez :



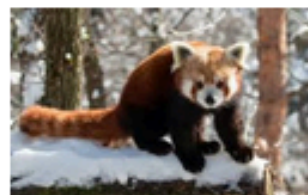
Drag and drop images, annotations, and videos.

 .jpg, .png, .bmp, .webp, .avif  in [26 formats](#)  .mov, .mp4
*Max size of 20MB and 16,400 × 10,900 pixels.

Select Files

Select Folder

Save and Continue →



9.jpeg



8.jpeg



7.jpeg



6.jpeg



5.jpeg



4.jpeg



3.jpeg



2.jpeg



10.jpeg



1.jpeg

PAS À PAS

Si vous annotez seul·e :

How do you want to label your images?

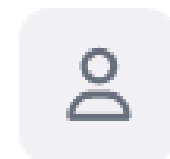


Auto-Label Entire Batch



Use your own custom model or a zero-shot model to automatically label your entire batch.

✦ Try with SAM3



Label Myself

Label images with our AI labeling tools.



Label With My Team



Split up the labeling work across your team.



Hire Outsourced Labelers

Trial

Work with an professional labeling team vetted by Roboflow.

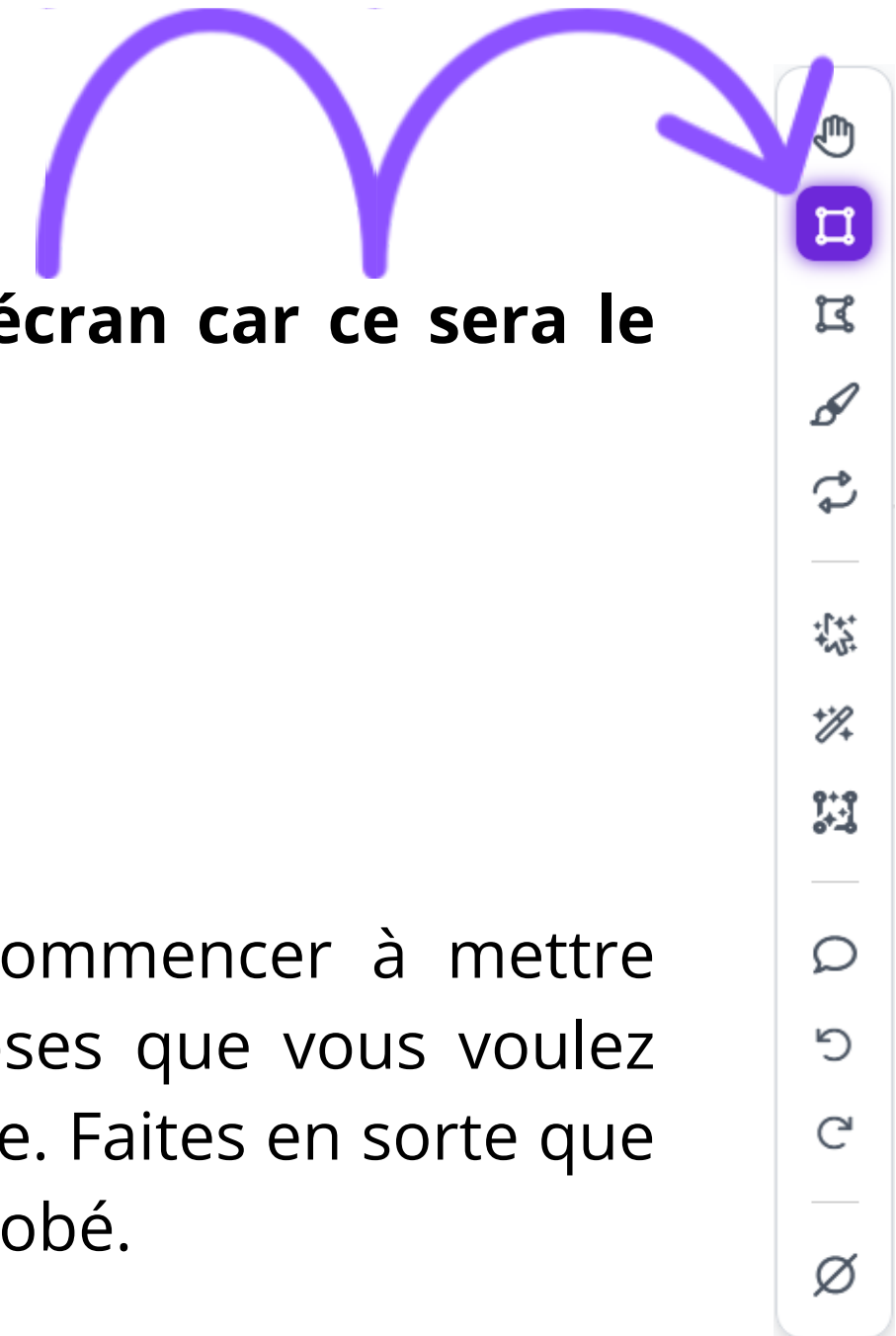
Si vous annotez à plusieurs :

ANNOTATIONS

Il faut bien sélectionner la "bounding box" dans l'outil à droite de l'écran car ce sera le format que YOLO comprendra.

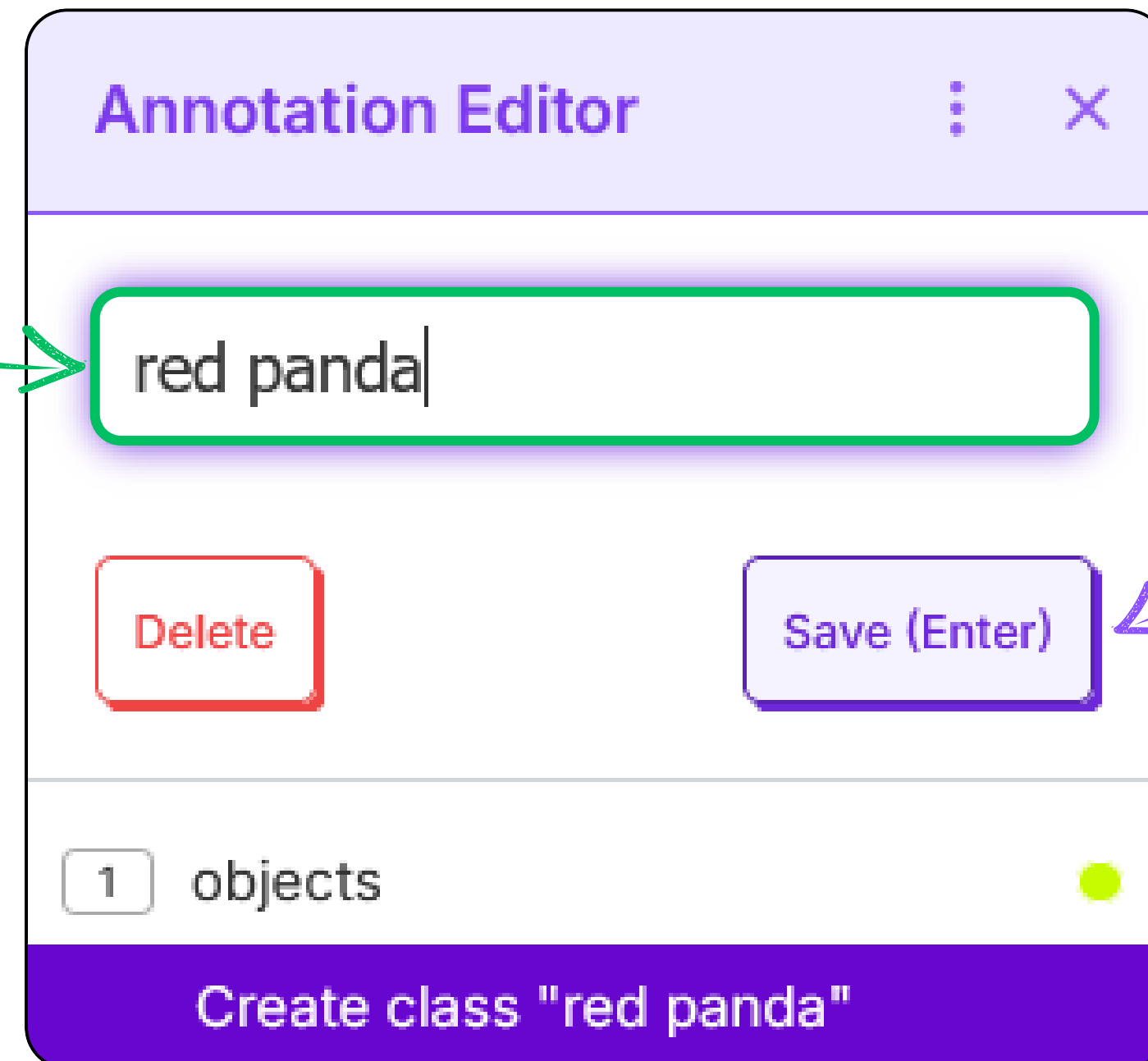


Ensuite vous pourrez commencer à mettre dans des boîtes les choses que vous voulez que votre modèle détecte. Faites en sorte que votre objet soit bien englobé.



ANNOTATIONS

Rentrez le nom de la classe de l'objet que vous venez d'encadrer ici :

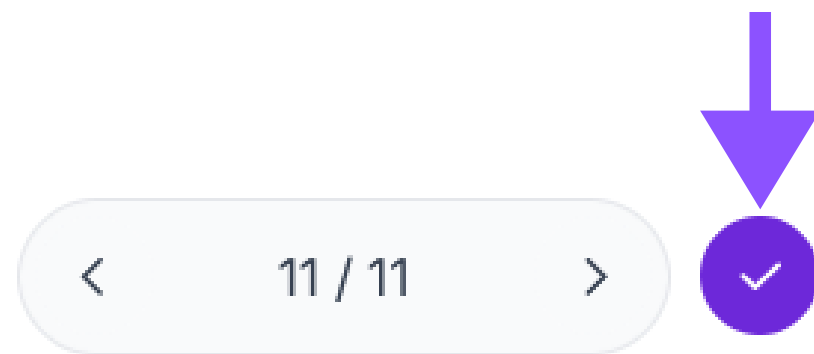
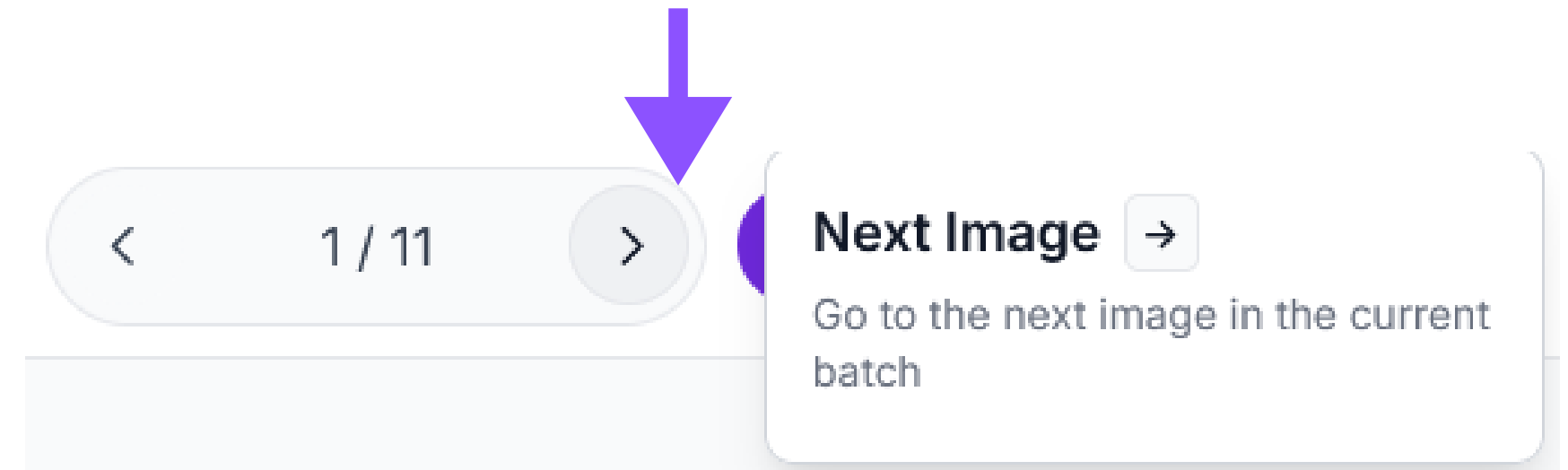


The image shows a screenshot of a software interface titled "Annotation Editor". At the top, there is a purple header bar with the title "Annotation Editor" and window control icons (three dots and a close 'X'). Below the header is a large text input field containing the text "red panda", which is highlighted with a green border. Underneath the input field are two buttons: a red "Delete" button on the left and a purple "Save (Enter)" button on the right. At the bottom of the editor, there is a status bar showing "1 objects" with a small green dot to its right. A dark purple footer bar at the very bottom contains the text "Create class 'red panda'".

Validez

ANNOTATION

Pour aller à l'image d'après, il suffit de cliquer sur la flèche droite en haut de la page.



Une fois toutes les annotations terminées, vous pouvez cliquer sur ce bouton pour ajouter les images annotées au dataset final.

Une fois l'annotation finie, vous avez une fenêtre qui apparait. Cette fenêtre vous permet de pouvoir choisir la répartition de vos images entre 3 groupes : train, val et test. Ces trois groupes ont des objectifs bien définis :

- train permet d'entraîner votre modèle (**ne peut pas être vide**)
- val permet que le modèle s'autoévalue de temps en temps lors de l'entraînement
- test permet d'évaluer les capacités d'un modèle entraîné

Il est important de mettre des images dans chaque dossier. Pour pouvez garder le découpage de Roboflow 70-20-10 qui est très bien (si vous n'en avez pas beaucoup, mettez en au moins dans val au détriment de test, mais **ne mettez jamais val vide**).

Add Images To Dataset

Labeled Images to Add: 10
Remaining Batch Images: 0

Your remaining images will be split into a new batch.

Method: Use Existing Values (What's Train, Valid, Test?)

Distribution: Images with an existing split (10)
Train: 10 images Valid: 0 images Test: 0 images

Add 10 Images

Tout dans "train"

Add Images To Dataset

Labeled Images to Add: 10
Remaining Batch Images: 0

Your remaining images will be split into a new batch.

Method: Split Images Between Train/Valid/Test (What's Train, Valid, Test?)

Distribution: Train 70%, Valid 20%, Test 10%

Train: 7 images Valid: 2 images Test: 1 images

Add 10 Images

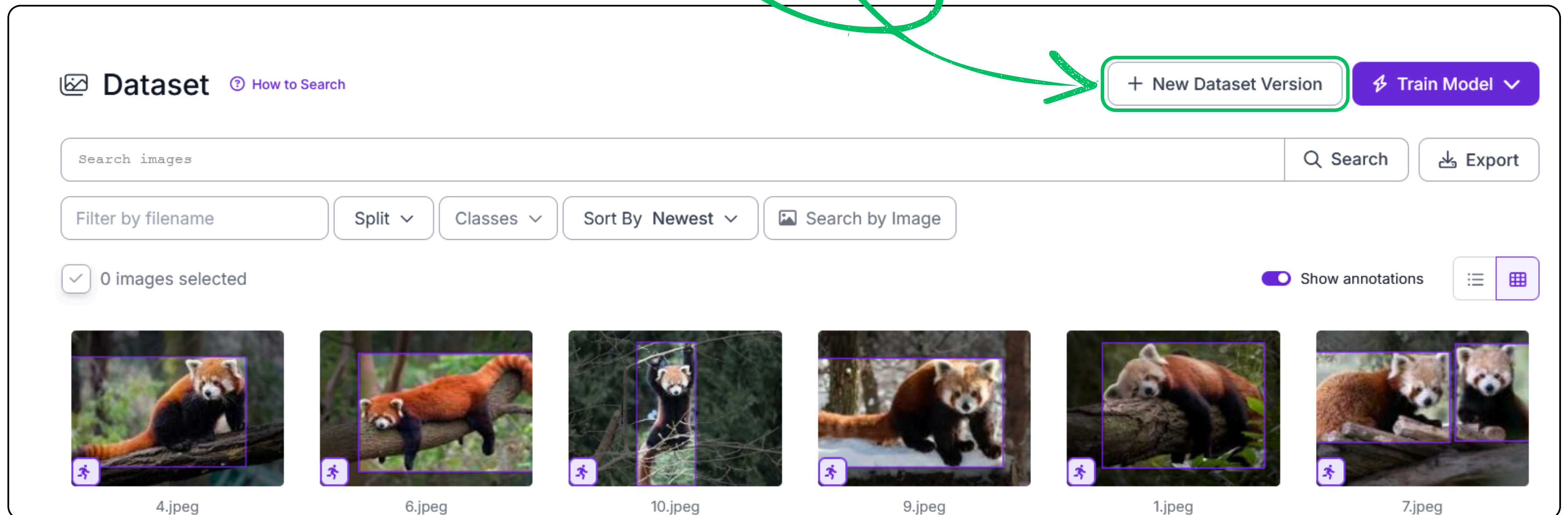
Choisissez en amont

Si vous avez envie de changer la répartition de vos images à posteriori, vous pouvez le faire ainsi : pour changer une image de groupe (passer de train à val par exemple), sélectionnez la puis dans l'onglet action cliquez sur "Change Dataset Split" et choisissez le split qui vous convient.

The screenshot displays the ISDM Dataset management interface. At the top, there are buttons for '+ New Dataset Version' and 'Train Model'. Below these is a search bar and an 'Export' button. A row of filters includes 'Filter by filename', 'Split', 'Classes', 'Sort By Newest', and 'Search by Image'. A notification bar indicates '1 image selected' and shows an 'Actions' dropdown menu. The menu options are: 'Add tags & metadata', 'Reassign For Labeling', 'Mark Null', 'Change Dataset Split', and 'Remove from Project'. The 'Change Dataset Split' option is highlighted. Below the menu, three image thumbnails are visible with their filenames: 'ibis falcinelle.JPG', 'chevalier gambette.JPG', and 'Fou de bassan (2).JPG'. An orange arrow points from the 'Change Dataset Split' option in the menu to the 'Train Model' button at the top right.

Si vous exportez maintenant grâce au bouton "Export", les données n'auront pas le bon format. **Nous devons forcément passer par l'étape de faire une *version* des données.**

Ce qui est pratique avec la concept de version, c'est qu'on pourra être amené·e à en faire plusieurs, vous comprendrez l'intérêt plus tard.



The screenshot shows a web interface for managing a dataset. At the top left, there is a 'Dataset' header with a 'How to Search' link. Below this is a search bar with the placeholder text 'Search images' and a 'Search' button. To the right of the search bar is an 'Export' button. Below the search bar are several filter and action buttons: 'Filter by filename', 'Split', 'Classes', 'Sort By Newest', and 'Search by Image'. On the left side, there is a selection indicator showing '0 images selected'. On the right side, there is a 'Show annotations' toggle switch and a view toggle (list/grid). A green arrow points from the text above to the '+ New Dataset Version' button, which is highlighted with a green border. Below the filters, there are six image thumbnails of red pandas, each with a bounding box and a small icon in the bottom left corner. The thumbnails are labeled with filenames: 4.jpeg, 6.jpeg, 10.jpeg, 9.jpeg, 1.jpeg, and 7.jpeg.

DATASET VERSION

Explications préalables au versionning

On a vu au tout début l'intérêt d'avoir un large éventail de données, cependant, ce n'est pas forcément aisé de rassembler plein d'images avec l'objet sous tous les angles, donc il est possible de traiter les images numériquement pour leur appliquer des variations.

Il y a deux types de modification des données : le **preprocessing** et l'**augmentation**. On va passer sur l'utilité et les subtilités de chacune des deux étapes.

PREPROCESSING

Le preprocessing est un traitement que l'on va appliquer à nos images pour diverses raisons.

Partons de nos images de bases (celle que l'on vient d'annoter), on va dire qu'on en a un nombre x .

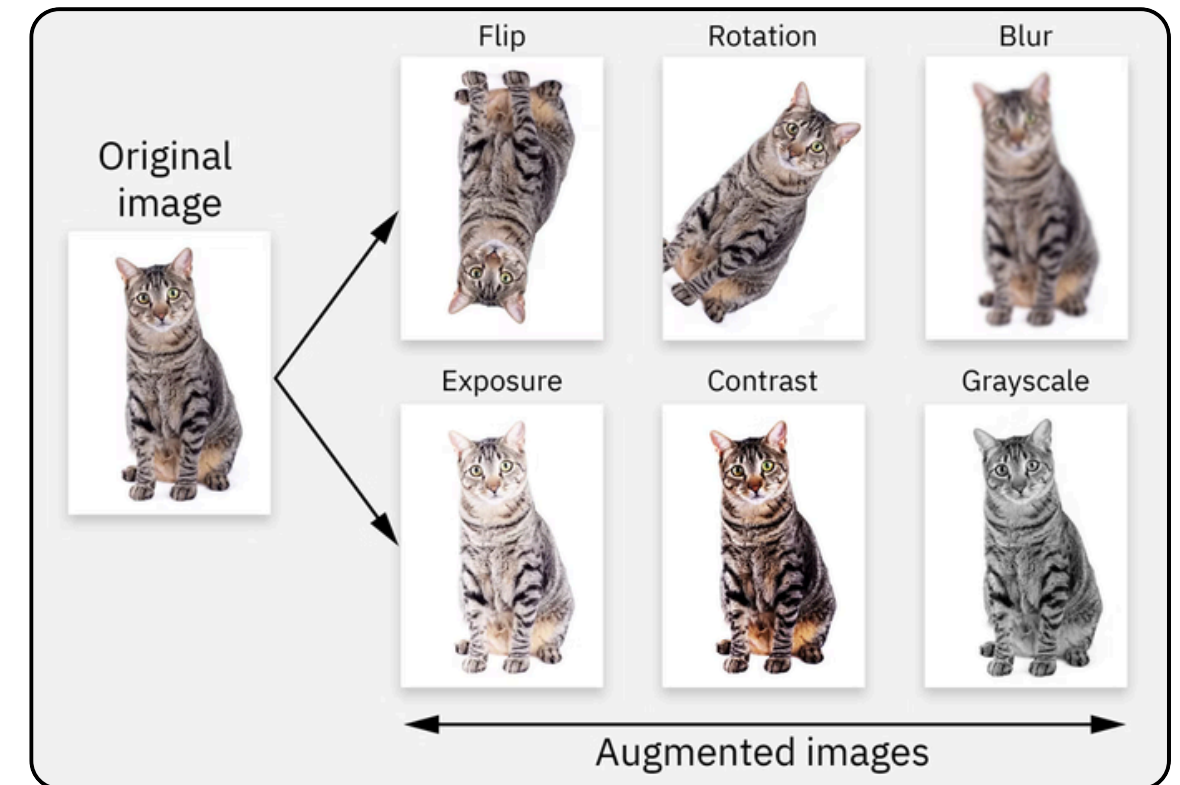
Le preprocessing s'applique à toutes les images de base donc avant le preprocessing on en a x et après aussi. **Le preprocessing ne change pas notre nombre d'image.** Cependant il modifie nos images de base donc **on "perd" nos images de base, vu qu'on ne garde que les modifiées.**

Quelques exemples de preprocessing et leur utilité :

- **ajustement de la taille** : très pertinent pour rendre uniforme notre dataset, on reviendra dessus.
- **conversion en niveau de gris** : si les couleurs ne sont pas pertinentes, convertir en nuances de gris permet de gagner de l'espace de stockage. *Attention, il ne restera que des images en nuances de gris, les images couleurs ne seront plus présentes.*

AUGMENTATION

L'augmentation, comme son nom l'indique va permettre de **grossir notre base de données**. L'intérêt principal est de varier les images comme le montre le schéma ci-contre : au lieu d'avoir juste une image d'un chat dans le "bon" sens et en couleur, on va appliquer des transformations, donc ici le nom au dessus de chaque image correspond à l'opération appliquée, **pour qu'ensuite le modèle soit capable de reconnaître un chat même si il est en nuance de gris et à l'envers**.



En sortie de l'augmentation on a donc **plus de données qu'en entrée**, comme dans l'exemple ci-dessus, si on prend x notre nombre d'image et qu'on applique les 6 transformations du schéma, on aura $x + 6*x$ images donc $7*x$ images (x images de base, plus 6 transformations par images donc $7x$ images).

Pour reprendre la slide précédente, **si on rajoute les nuances de gris lors de l'augmentation**, on aura 2 fois les images de base avec x images colorés et x images en niveaux de gris ($2x$ images), donc l'entraînement se fera sur les deux (**à l'inverse du preprocessing où les images colorées disparaissent**).

DATASET VERSION

Informations préalables

Selon votre cas, vous pouvez choisir de faire du preprocessing et/ou de l'augmentation de données comme nous avons vu. Mais ces changements peuvent soit se faire soit *en amont*, soit *pendant* l'entraînement.

Si ces changements sont faits en amont : toutes les variations de l'image principales seront *créées* donc potentiellement beaucoup de place sera utilisée, mais le modèle n'aura plus qu'à les prendre, donc l'entraînement sera plus rapide. **Les changements *en amont* sont fait sur *Roboflow*.**

À l'inverse, si ces changements sont faits pendant l'entraînement : toutes les variations de l'image principales seront créées en direct et ne seront stockées que temporairement, mais cela prendra du temps de calcul au modèle qui devra fabriquer les images en cours de route. **Les changements *pendant* sont faits par *YOLOv26*** (*Yolov26 est le modèle d'intelligence artificielle que nous utiliserons tout à l'heure*). À savoir que YOLO a des paramètres d'augmentation par défaut, donc l'augmentation peut se faire automatiquement sans avoir besoin de les paramétrer.

DATASET VERSION

Informations préalables

Sur Roboflow, vous pouvez voir exactement ce qui est fait, et choisir point par point ce que vous voulez faire. En revanche, YOLOv26 applique beaucoup de transformation par défaut, si on veut les désactiver il faut le faire à la main pour chaque transformation, ce qui peut s'avérer un peu fastidieux.

Donc ***si vous êtes débutant***, il est recommandé de ne faire aucune augmentation sur Roboflow car YOLO s'en occupe déjà (vous pouvez quand même faire du preprocessing selon vos besoins). Vous n'avez donc pas besoin de vous embarquer dans la compréhension de chaque transformation.

Sinon, le principal argument qui permet de choisir comment faire l'augmentation c'est la **ressource de calcul**. Si vous en avez beaucoup (vous travaillez sur le cluster par exemple), vous pouvez laisser YOLO faire plus de calculs, mais si vous êtes sur votre machine personnel sans trop de ressources, préférez faire l'augmentation sur Roboflow, cela prendra plus de place mais rendra l'entraînement plus rapide. Dans ce dernier cas, n'oubliez pas de désactiver les nombreux paramètres d'augmentation de YOLO pour ne pas avoir des données complètement difforme à cause de transformations combinées (le script qui permet de tout désactiver vous sera fourni en temps voulu).

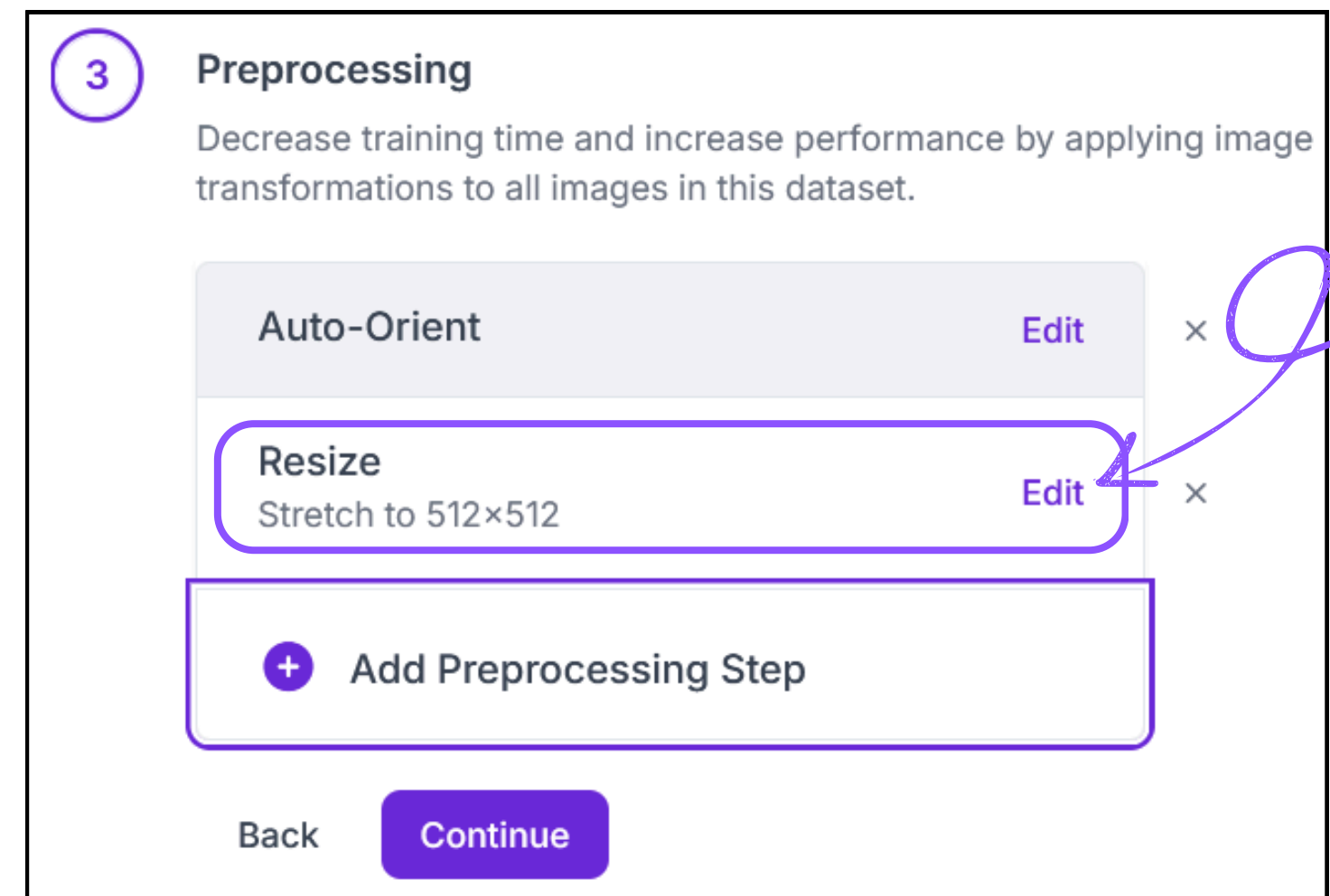
DATASET VERSION

Preprocessing

Maintenant, la pratique ! Vous pouvez modifier et ajouter des étapes de preprocessing à appliquer à vos données, sachant que l'auto-orient et le resize sont appliqués automatiquement ce qui ne veut pas dire qu'ils sont obligatoires, vous pouvez totalement les enlever ou les modifier.

À savoir que Yolo a besoin d'images carrées dont la taille est un multiple de 32 (pour des raisons de découpage), donc les tailles standards sont 640 ou 1280, mais vous pouvez mettre celle qui vous arrange. Si vous mettez une taille qui n'est pas multiple de 32, Yolo arrondira à la valeur la plus proche.

Par contre, allez dans le "Edit" du resize pour choisir la façon dont l'image est déformée (vous serez guidé·e par Roboflow).



DATASET VERSION

Preprocessing

Pour chaque étape de preprocessing, il y aura un **lien vers une documentation spécifique** à la transformation. Cela vous permet de vous renseigner avant d'appliquer des changements.

Certaines opérations de preprocessing sont réservées aux personnes qui paient un abonnement donc à moins que vous ne soyez dans vos deux semaines gratuites, tout ne sera pas disponible.

Auto-Adjust Contrast

Boosts contrast based on the image's histogram to improve normalization and line detection in varying lighting conditions.

Type: Adaptive Equalization

original

adjusted

Why should I use contrast as a preprocessing step? [↗](#)

To help your model better detect edges.

via Roboflow Blog

Go Back Apply

Vous avez aussi la documentation sur le preprocessing de manière générale ici :

<https://docs.roboflow.com/datasets/dataset-versions/image-preprocessing>

AUGMENTATION DE DONNÉES

Une fois toutes vos étapes de preprocessing ajoutées, appuyez sur “**Continue**” et vous serez amené·e à l'étape de l'augmentation des données.

Pour rappel, **le plus simple est de ne rien mettre à cette étape** et de laisser YOLO gérer tout seul l'augmentation. Vous pouvez néanmoins regarder ce qui est proposé car cela peut vous donner des exemples visuels de plusieurs types d'augmentation.

Comme pour le preprocessing, certaines augmentations sont pour les abonné·es.

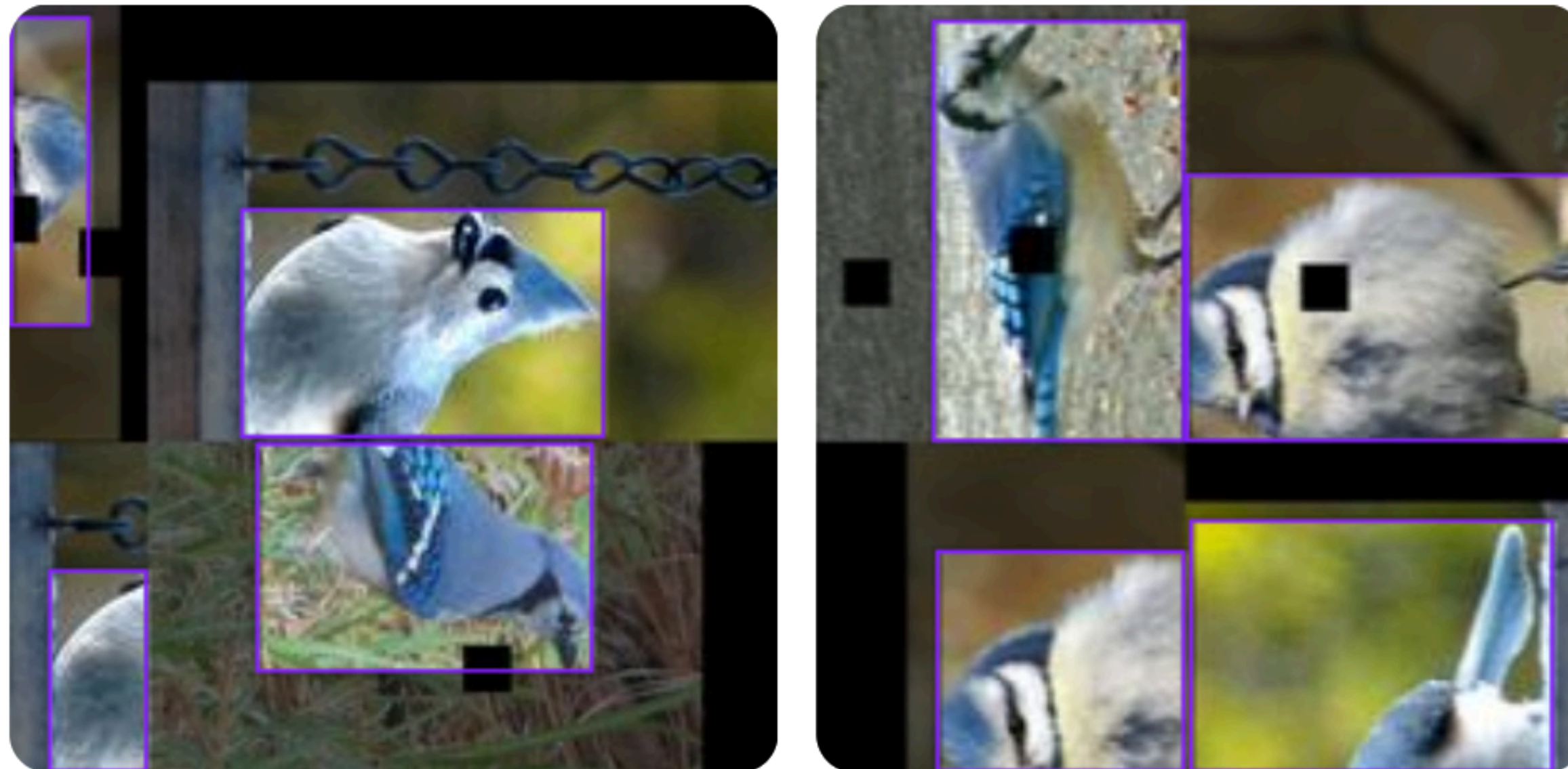
La documentation de Roboflow concernant l'augmentation de données :

<https://docs.roboflow.com/datasets/dataset-versions/image-augmentation>

AUGMENTATION DE DONNÉES

Exemple

Dans mon jeu de données d'oiseaux bleus, j'ai choisi comme augmentation de la *mosaïque* et de la *rotation* dans trois sens. Roboflow les combine et cela donne ça :

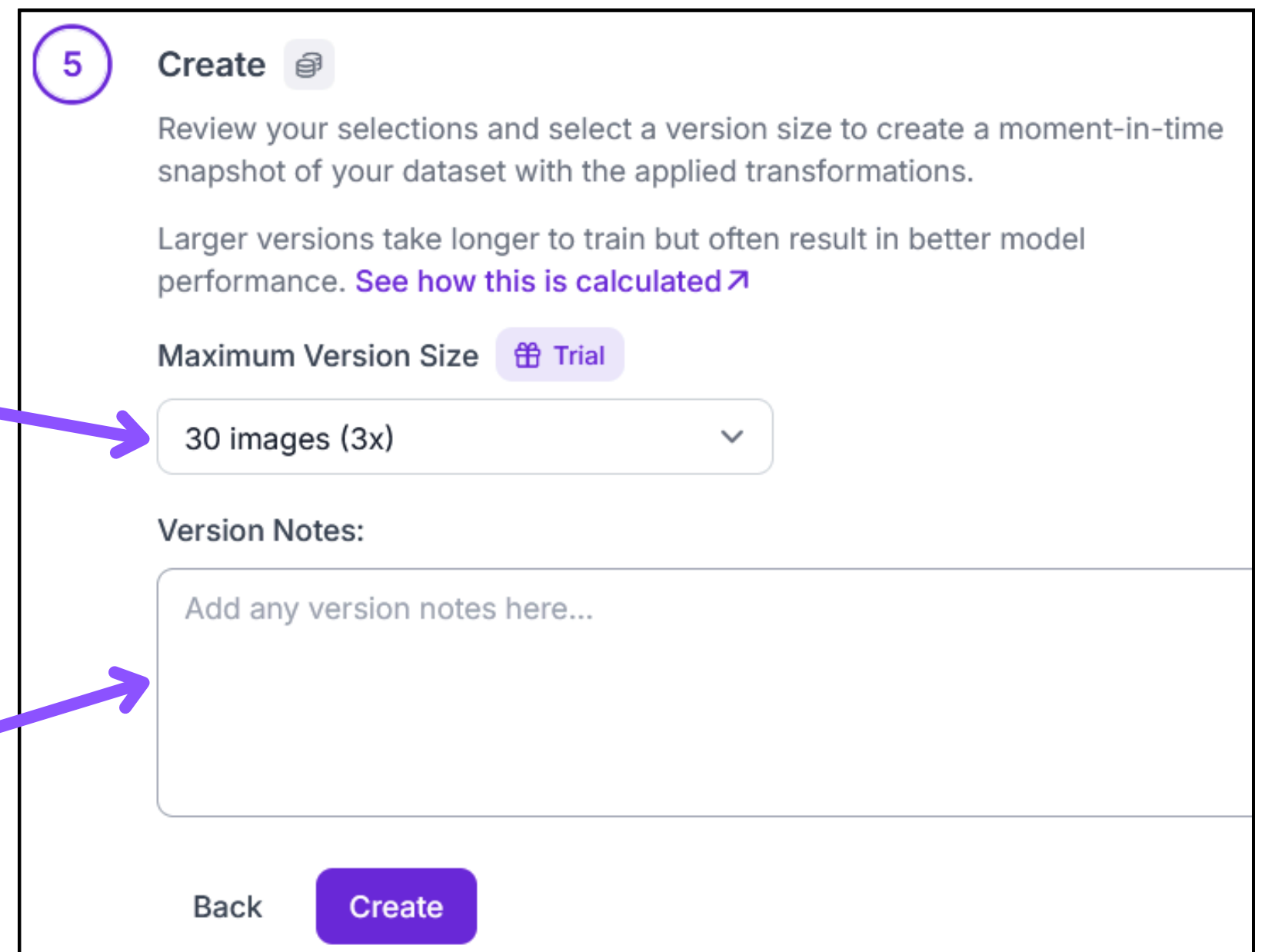



DERNIÈRE ÉTAPE “CREATE”

Enfin, l'étape 5 "Create" permet de conclure le versionning.

Si vous avez fait de l'augmentation, vous pouvez **choisir combien d'images vous voulez** que Roboflow farbique à partir de vos images de base.


Et vous pouvez **rajouter des notes** à la fin. Cela peut-être pertinent si vous comptez tester plusieurs combinaisons preprocessing/augments.




5 Create 

Review your selections and select a version size to create a moment-in-time snapshot of your dataset with the applied transformations.


Larger versions take longer to train but often result in better model performance. [See how this is calculated ↗](#)

Maximum Version Size  Trial

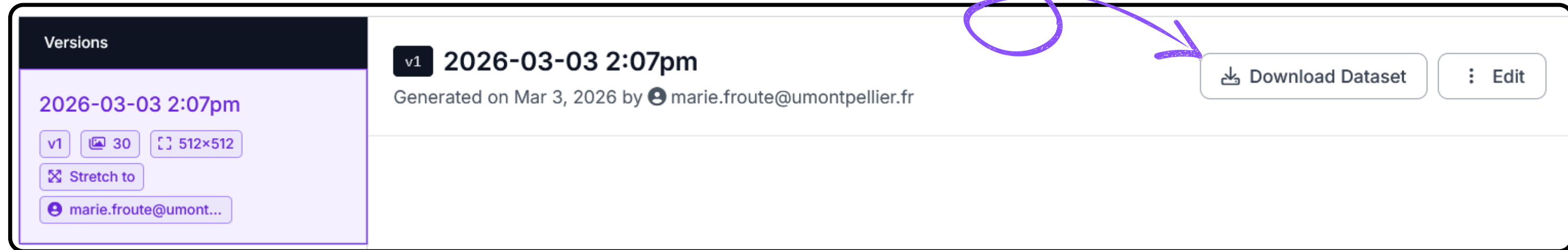
30 images (3x) 

Version Notes:


Add any version notes here...

Back 



EXPORTATION DES DONNÉES





Versions



v1 2026-03-03 2:07pm
Generated on Mar 3, 2026 by  marie.froute@umontpellier.fr

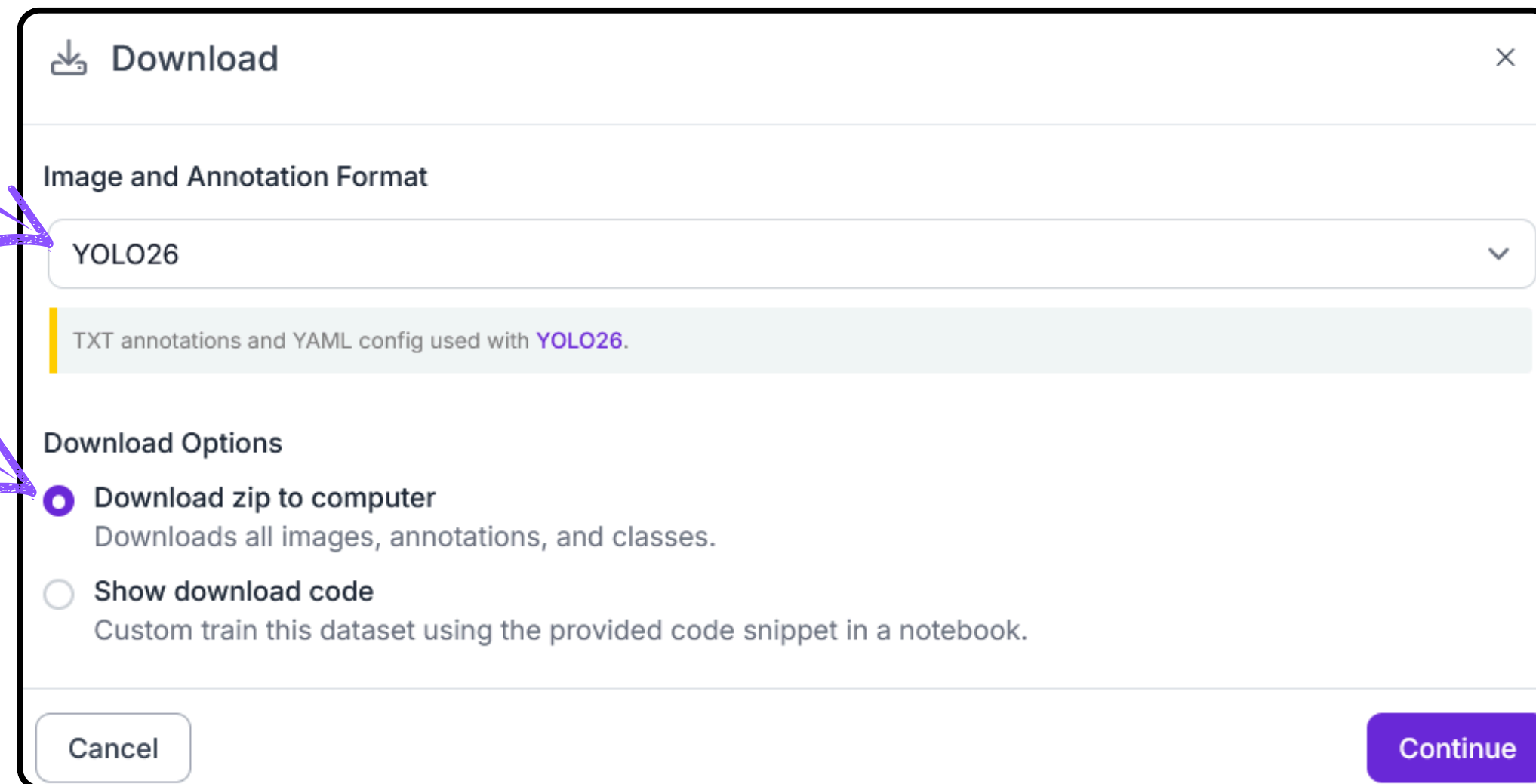
2026-03-03 2:07pm

v1  30  512x512

 Stretch to

 marie.froute@umont...

 Download Dataset  Edit





 Download ✕

Image and Annotation Format

YOLO26 

TXT annotations and YAML config used with [YOLO26](#).

Download Options

Download zip to computer
Downloads all images, annotations, and classes.

Show download code
Custom train this dataset using the provided code snippet in a notebook.

Cancel Continue

EXPORTATION DES DONNÉES

Un fois le .zip téléchargé, extrayez le dans le **dossier dans lequel vous travaillerez**.

Vous devriez avoir plusieurs choses dedans :

- fichier **README roboflow.txt** : contient les métadonnées de roboflow
- dossiers **train, valid, test** : sont là en fonction de ce que vous avez choisi dans Roboflow (c'est possible que vous n'ayez que le fichier "train", c'est normal)
- fichier **data.yaml** : fichier important pour Yolo, va lui permettre de trouver les données

MISE EN FORME DES DONNÉES

Ouvrez le fichier .yaml (à l'aide de n'importe quel outil d'édition de texte), il devrait ressembler à celui ci-contre (avec en plus un paragraphe Roboflow non montré ici).

Trois lignes avec les **chemins relatifs** vers les différents dossiers d'entraînement, de validation et de test (si vous rencontrez des problèmes liés à ça et que vous travaillez en local, n'hésitez pas à mettre les chemins absolus, mais ne pas faire ça si vous comptez travailler sur un cluster). **Il doit y avoir les trois chemins vers train, val ET test même si vous n'avez pas les trois dossiers d'images.**

Il faudra que votre fichier python soit dans le même dossier.

Un paragraphe dédié aux **nomx des différentes classes** à détecter.

Ce fichier n'a normalement pas besoin d'être modifié, mais si vous rencontrez des problèmes, vous trouverez plein de ressources sur internet car pas mal de monde a déjà éprouvé le modèle.

```
train: ../train/images
val: ../valid/images
test: ../test/images

nc: 1
names: ['Blue jay']
```