

Projet Réserve : matériel typographique

Guide pratique (MacOSX ou Linux)

Plan

- A. Obtenir les images à traiter*
- B. Entraîner un modèle*
- C. Obtenir les annotations réalisées dans Roboflow et les visualiser dans Panoptic*
- D. Utiliser le modèle Roboflow pour traiter d'autres documents*
- E. Convertir les annotations vers le standard IIIF*

A. Obtenir les images des documents à traiter















Les images des documents sont obtenues grâce à l'API IIIF Image de Gallica. En entrée du script `extract_iiif.py`, il faut fournir une liste d'ARK de documents Gallica :

```
ark:/12148/btv1b86000632
ark:/12148/btv1b22000344
ark:/12148/bpt6k313285x
...
```

Et le ratio relatif à la taille d'image souhaitée (> 0.0 et ≤ 1.0) puis lancer le script :

```
>python extract_iiif.py arks.txt 0.7
```

Les images sont enregistrées dans un dossier `IIIF_images`, dans des sous-dossiers nommés d'après l'ARK des documents Gallica. En cas d'échec du téléchargement de certaines images, il suffit de relancer le script (les images déjà téléchargées ne seront pas concernées).

output	
Nom	
▼	btv1b86000632
	btv1b86000632-0001.jpg
	btv1b86000632-0002.jpg
	btv1b86000632-0003.jpg
	btv1b86000632-0004.jpg
	btv1b86000632-0005.jpg
	btv1b86000632-0006.jpg
	btv1b86000632-0007.jpg
	btv1b86000632-0008.jpg
	btv1b86000632-0009.jpg
	btv1b86000632-0010.jpg
	btv1b86000632-0011.jpg
	btv1b86000632-0012.jpg
	btv1b86000632-0013.jpg
	btv1b86000632-0014.jpg
178 éléments, 241,69 Go disponible(s)	

Notes :

- Le ratio de taille image extraite/image scannée doit être mémorisé pour les étapes ultérieures (par ex. 70 %)
- Le lien avec les documents Gallica est préservé en nommant les fichiers image d'après le schéma `ark-vue`

B. Entraîner un modèle Roboflow

Méthodologie :

1. Sélectionner un corpus d'annotation
2. Ingérer les images du corpus dans Roboflow
3. Annoter le corpus
4. Entraîner le modèle dans Roboflow

Voir :

- guide pratique Roboflow
- présentation technique du projet SnoopTypo

C. Obtenir les annotations réalisées dans Roboflow et les visualiser dans Panoptic

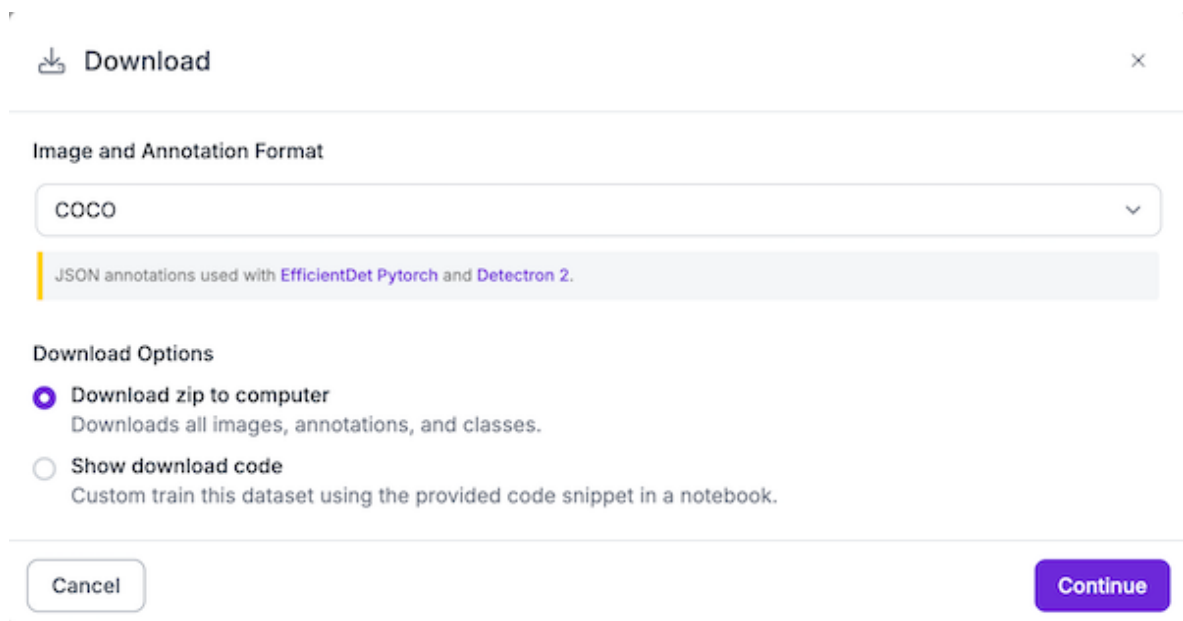
Ce processus permet de visualiser les éléments précédemment annotés dans Roboflow avec l'outil Panoptic, de corriger éventuellement certaines annotations, d'en ajouter de nouvelles, d'ouvrir les documents annotés dans Gallica.

Cette phase permet d'appréhender les données annotées mais aussi de les réutiliser dans les mêmes conditions que les données produites par inférence (étape D).

1. Exporter le jeu de données annoté depuis Roboflow, au format COCO

Dans Roboflow : <https://app.roboflow.com/snooptypo/snooptypo/models>








- onglet Versions
- bouton Download Dataset
- Format : COCO, option : Download zip to computer



The screenshot shows a 'Download' modal window from Roboflow. At the top, there's a 'Download' button with a download icon and a close 'x' button. Below this, the 'Image and Annotation Format' is set to 'COCO'. A note indicates 'JSON annotations used with EfficientDet Pytorch and Detectron 2.' Under 'Download Options', the 'Download zip to computer' option is selected with a radio button, and a description says 'Downloads all images, annotations, and classes.' The 'Show download code' option is also present with a description 'Custom train this dataset using the provided code snippet in a notebook.' At the bottom, there are 'Cancel' and 'Continue' buttons.






ATTENTION :

Le jeu de données doit avoir été généré **sans augmentation**, dans Roboflow, sinon les mêmes images seront présentes plusieurs fois. Ici, un jeu a été produit à cet effet avec le bouton « New Version » (version 2025-06-30), sans duplication (5830 images). Le jeu qui a été utilisé pour l'entraînement du modèle est la version (2024-11-13, 14514 images).

Versions	
2025-06-30 12:05pm	
v3	 5830
 jean-philippe Moreux	
2024-11-13 1:23pm	
v2	 14514  Accurate
 COCOs	
 jean-philippe Moreux	

2. Préparer le traitement

- Dézipper le .zip dans le dossier de traitement. Les images annotées sont réparties dans 3 sous-dossiers : test, train, valid

Nom
 README.dataset.txt
 README.roboflow.txt
>  test
>  train
>  valid

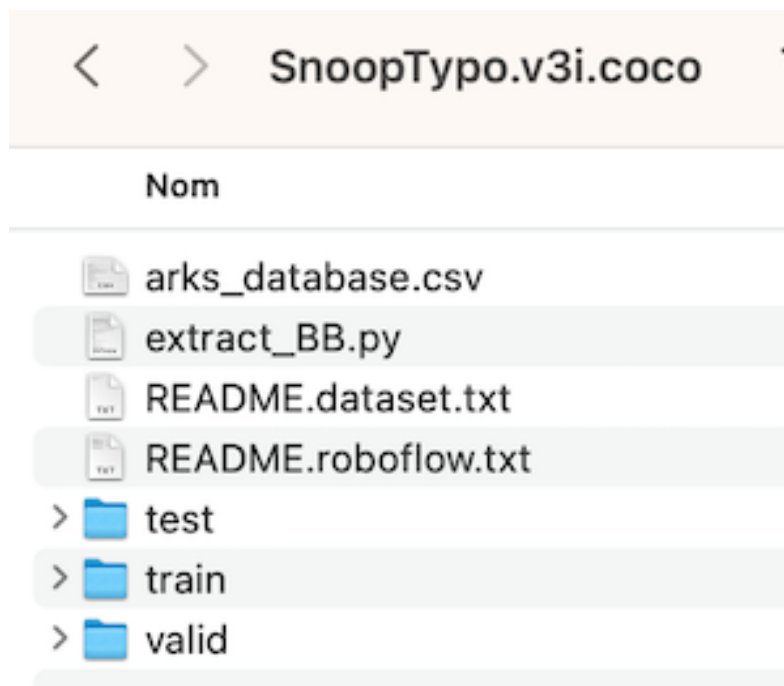
Chaque sous-dossier contient un fichier JSON (données des annotations) et les

images annotées.

- Le dossier de traitement doit contenir la liste des ARK des documents utilisés dans le jeu de données, afin de rétablir le lien vers Gallica. Ce lien est réalisé d'après le titre du document, qui est porté dans le nom des fichiers images. Ces données sont contenues dans le fichier `arks_database.csv` :

```
Title#ARK
[Illustrations de De la Déclaration des louenges de follies] /
[Non identifié] ; Erasme, aut. du texte#ark:/12148/
btv1b22000344
[Le Livre de François Patrice de l'Institution et
administration de la chose publique traduit de latin en
françois.]#ark:/12148/bpt6k313285x
...
```

- Le script de traitement `extract_boxes.py` doit être copié dans le dossier du jeu de données. Au final, le dossier doit ressembler à :



3. Traiter les données annotées

Ce script traite les annotations JSON COCO afin d'extraire les cadres des contenus annotés (lettres ornées, ornements, etc.), de les superposer sur les images et de générer des imagelettes des contenus. Il génère aussi des données CSV pour un traitement ultérieur ainsi que les données CSV qui seront nécessaires à l'importation dans Panoptic de métadonnées utiles (notamment le lien URL vers Gallica).

Les imagelettes produites sont de deux natures :

- extraite des images de Roboflow,
- générée via l'API Gallica IIIF (à la meilleure résolution disponible) : en option, doit être demandé lors de l'appel avec `-i`

Le script doit être lancé sur chaque sous-dossier `test`, `train`, `valid`. Exemple avec le dossier `test` :

```
> python extract_boxes.py test 0.7 -i
```

Après traitement, les données produites sont stockées dans un dossier `output` :

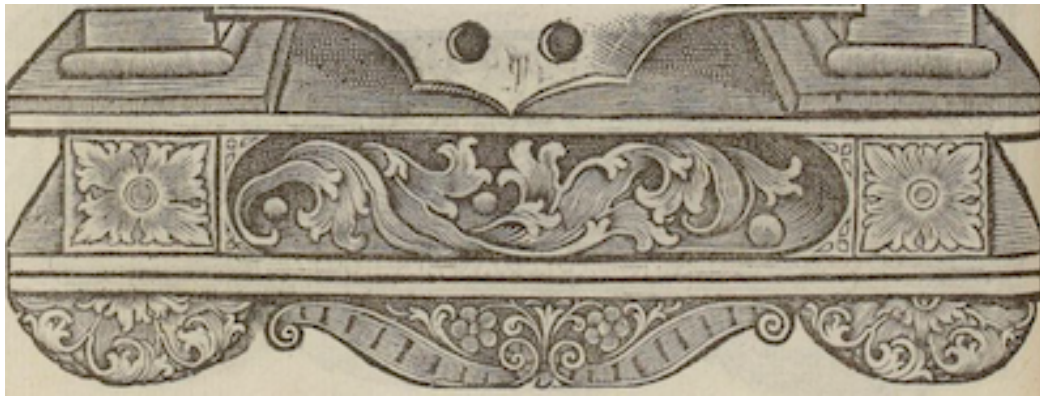
- images des pages avec les boîtes des contenus annotés en surimpression
- imagerie IIIF des contenus (dossier `IIIF_thumbs`), organisées par ARK
- imagerie extraites (dossier `thumbs`), organisées par ARK et type de contenus
- données CSV :
 - `processed_data.csv` : une ligne par annotation
`ARK,Vue,Image_filename,Category_name,Gallica,IIIF,Annotation_filename`
 - `import_pano.csv` : une ligne par image annotée, pour importation ultérieure dans Panoptic
`path;Gallica[url];IIIF[url];Classe[tag];ARK[text]`
 - un fichier JSON par image annotée au format Roboflow Supervision, dans le dossier `SV`

Image annotée avec les boîtes englobantes :



Ces_presentes_Heures_a_lusaige_de_view_26_num_NP.jpg

Imagette de contenu (ici un ornement) :



Ces_presentes_Heures_a_lusaige_de_view_26_num_NP-Ornement_84

4. Importer le corpus dans Panoptic

L'importation de corpus iconographiques se fait en deux temps :

1. Importation des imagerie IIIIF
2. Importation des métadonnées associées (en option)

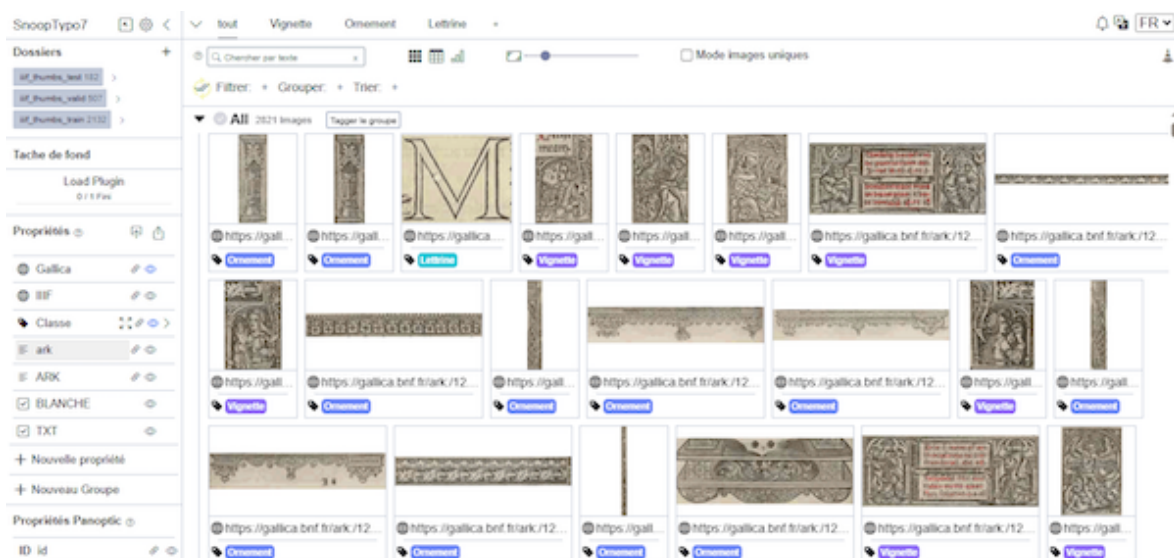
Pour un sous-dossier traité à l'étape précédente, il faut donc :

- a. Zipper les imagerie IIIIF contenues dans le dossier `output/IIIF_thumbs`
- b. Copier ce zip sur le serveur BnF accessible à Panoptic et le dézipper : `\num_datasets.bnf.fr\Num_Datasets$\`
- c. Importer ce dossier dans un projet Panoptic
- d. Une fois l'importation réalisée, importer les métadonnées (fichier `processed_data.csv`) dans Panoptic

Pour ces opérations, voir le guide pratique Panoptic : BnF-ADM-2025-046350-01

A ce stade, Panoptic doit afficher les imagerie et leurs métadonnées :

- URL vers Gallica (actionnable par Ctrl-clic)
- type (ornement, lettre, vignette)



D. Utiliser le modèle Roboflow pour traiter d'autres documents

Le modèle de segmentation entraîné dans Roboflow peut être utilisé pour annoter automatiquement de nouveaux documents (mode *inférence*).

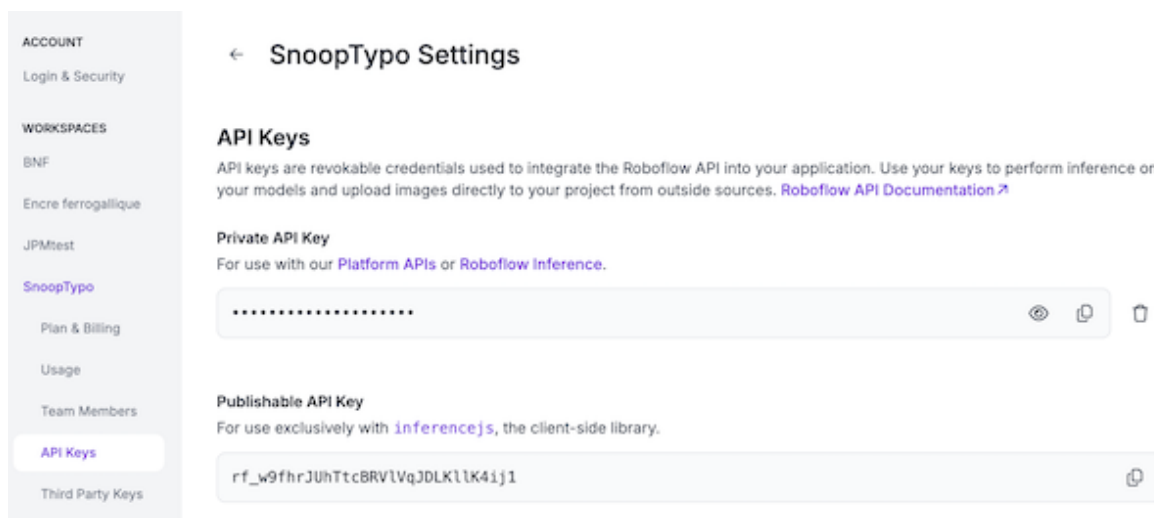
1. Installer l'environnement Python Roboflow

Dans un Terminal, créer un environnement Python :

```
>python -m venv roboflow  
>source roboflow/bin/activate  
>pip install inference
```

2. Identifier les descripteurs du modèle

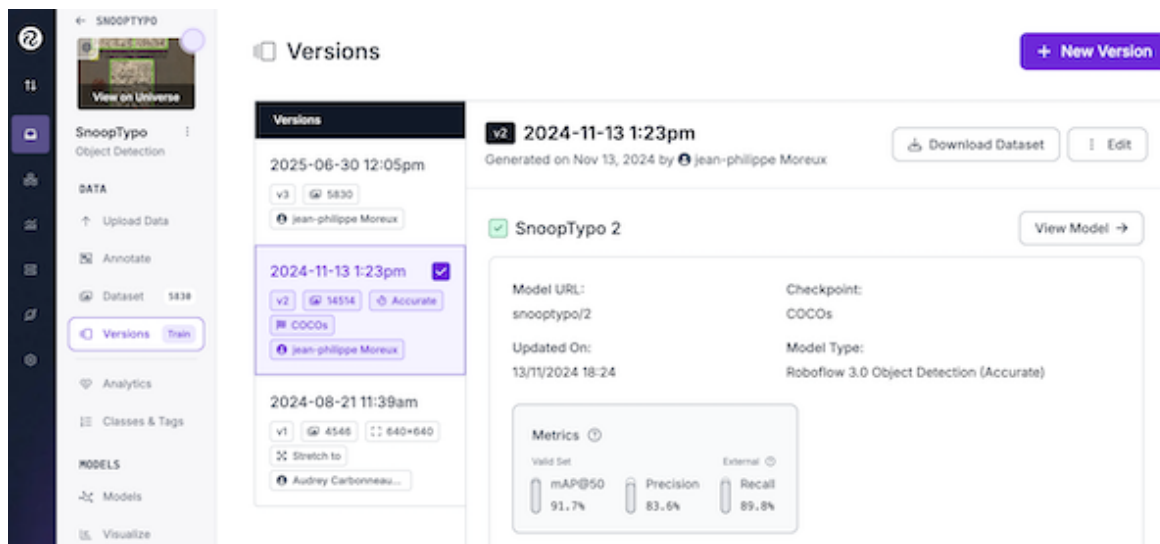
Identifier la clé API Roboflow du projet depuis le site roboflow, dans l'onglet Settings, puis « API Keys » :



La clé (« Private API Key ») doit être copiée puis collée dans cette ligne de commande :

```
>export ROBOFLOW_API_KEY="votre clé"
```

Identifier le nom du modèle qui a été entraîné dans Roboflow, dans l'onglet Versions du projet, puis en sélectionnant la version concernée. Le nom du modèle est donné dans le champs « Model URL », ici `snooptypo/2` :



3. Lancer les traitements

Pour lancer le traitement d'un dossier d'images d'un document Gallica, à l'aide d'un modèle nommée `snooptypo/2`, saisir cette commande dans le Terminal :

```
>python roboflow_inference.py btv1b86000632 snooptypo/2 -i
```

Le dossier doit être nommé d'après l'identifiant ARK. Le traitement récursif d'un dossier de dossiers est possible.

```
>python roboflow_inference.py IIIF_images/btv1b86000632
snooptypo/2 -i
```

Les résultats sont stockés dans un dossier JSON (un fichier de données par image, nommage `ark/ark-vue.json`).

Les éléments détectés dans les images sont décrits dans les données JSON : position dans l'image, type, confiance de la détection.

```
[
  {
    "x_min": 220.0,
    "y_min": 306.0,
    "x_max": 375.0,
    "y_max": 460.0,
    "class_id": 0,
    "confidence": 0.90966796875,
    "tracker_id": "",
    "class_name": "Lettrine",
    "file": "btv1b86000632/btv1b86000632-0001.jpg",
    "model": "snooptypo/2"
  }
]
```

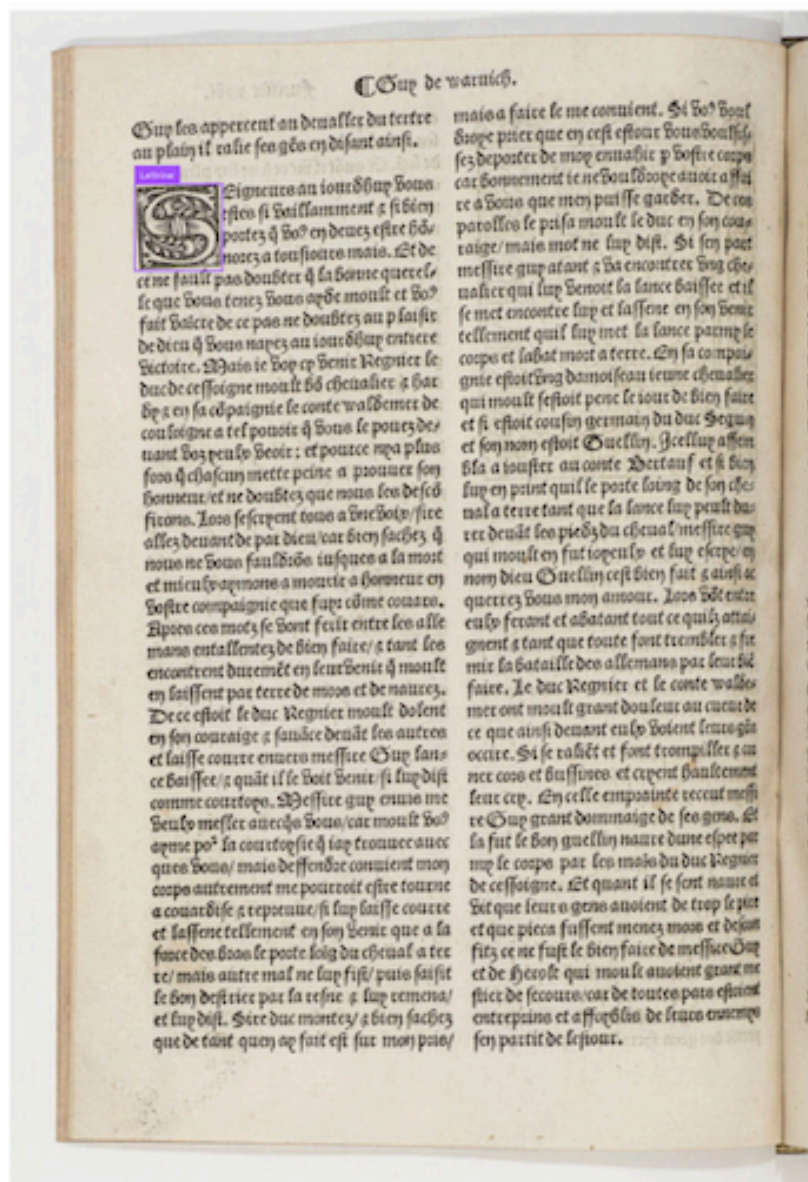
1

Avec l'option -s, les éléments détectés sont annotées sur l'image et cette dernière est sauvegardée (dans le dossier source) .

Avec l'option -i, des imageries IIIF des éléments détectés sont exportées via l'API Gallica IIIF.

Avec l'option -d, les éléments détectés sont annotées sur l'image et cette dernière est affichée :

Figure 1



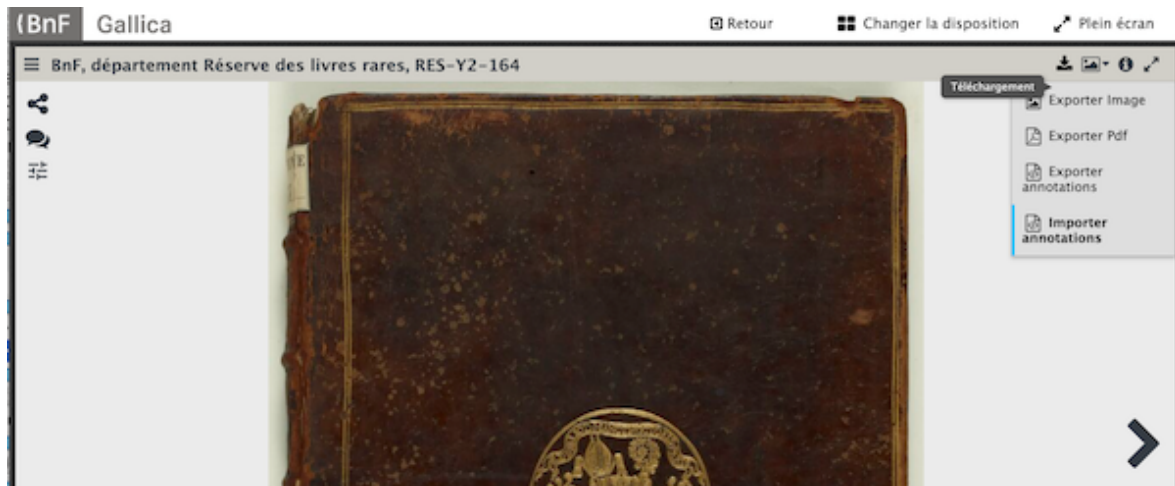
C. Convertir les annotations vers le standard IIIF

Les annotations JSON produites par l'étape précédente d'inférence peuvent être converties dans le standard IIIF afin d'être rendues exploitables par les outils compatibles IIIF. Le script `roboflow2iiif.py` réalise cette conversion pour un dossier de fichiers JSON produits par l'inférence Roboflow. Ce dossier doit être nommé d'après l'ARK du document Gallica. Le second paramètre du script est le ratio de taille utilisé lors de l'obtention des images via l'API IIIF (cf. section B.2)

```
>python roboflow2iiif.py JSON/btv1b86000632 0.7
```

Le script produit un fichier d'annotations IIIF (version IIIF Presentation 2.0) dans le dossier `IIIF_annotations`. Ce fichier peut ensuite être ouvert dans un visualiseur compatible IIIF :

1. Ouvrir le document dans un visualiseur IIIF, par exemple depuis Gallica (Mirador) : <https://gallica.bnf.fr/view3if/ga/ark:/12148/btv1b86000632>
2. Charger le fichier d'annotations JSON dans le visualiseur, via la fonction Téléchargements/Importer annotations.



3. Afficher les annotations, avec l'icône Annoter/voir les annotations :



Les annotations des éléments détectées sont alors visibles (avec leur type et le taux de confiance).

[✎ Editor](#) [🗑 Supprimer](#)

Vignette (0.93)



plat supéri...



contreplat ...



page de g...



page de g...



NP



NP



NP



NP

