

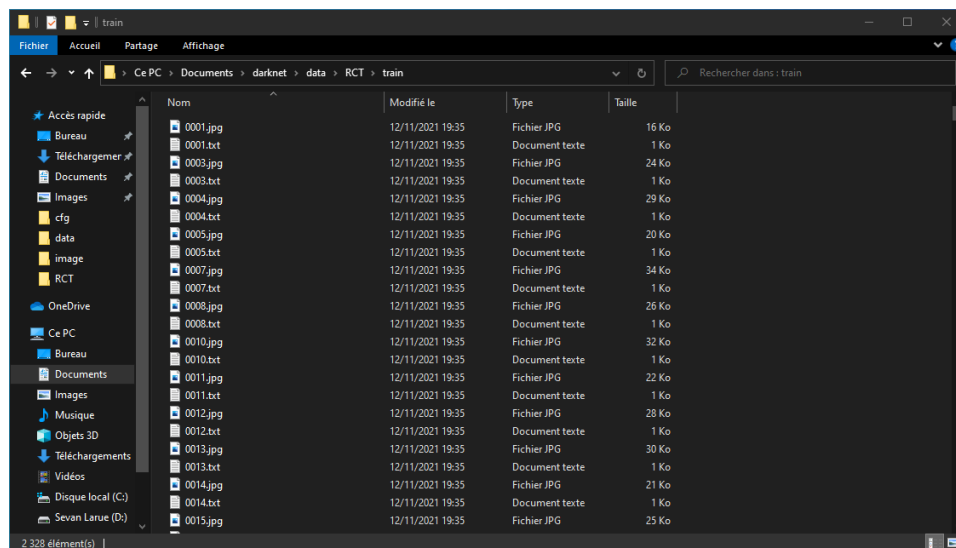
Entraîner un model YoloV3 et YoloV4

Source : <https://github.com/AlexeyAB/darknet#how-to-train-to-detect-your-custom-objects>

Tous les chemins sont donnés par rapport au fichier darknet.exe

Datasets

Mettre toutes les images au format .jpg dans un dossier au nom de votre choix et mettre ce dossier dans le dossier suivant : /data/VotreDataset



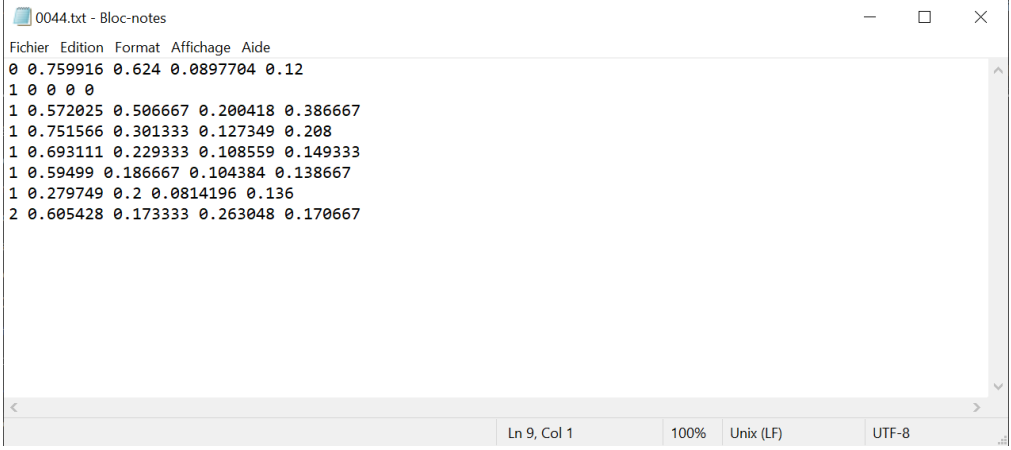
Chaque image doit être accompagné d'un fichier avec le même nom mais au format .txt. Ce fichier doit contenir les informations sur les objets présent dans l'image en suivant la norme suivante :

$\langle ObjectClasse \rangle \ \langle x_{centre} \rangle \ \langle y_{centre} \rangle \ \langle Width \rangle \ \langle Height \rangle$

Où :

- $\langle objectClass \rangle$ est un nombre entier correspondant à la classe de l'objet (allant de 0 à NombreDeClasse - 1)

- `<x_center>` `<y_center>` `<width>` `<height>` sont des nombre décimaux relatif à la taille de l'image. Ils sont compris entre 0,0 et 1,0.



```

Fichier Edition Format Affichage Aide
0 0.759916 0.624 0.0897704 0.12
1 0 0 0 0
1 0.572025 0.506667 0.200418 0.386667
1 0.751566 0.301333 0.127349 0.208
1 0.693111 0.229333 0.108559 0.149333
1 0.59499 0.186667 0.104384 0.138667
1 0.279749 0.2 0.0814196 0.136
2 0.605428 0.173333 0.263048 0.170667

```

Ln 9, Col 1 100% Unix (LF) UTF-8

Créer un fichier train.txt et val.txt contenant les noms et les chemins d'accès vers les images d'entraînement et de validation. Chaque ligne des ces fichiers correspond à une image différent et les chemin son relatif au fichiers darknet.exe. Les fichier train.txt et val.txt doivent être placer dans le dossier /data.



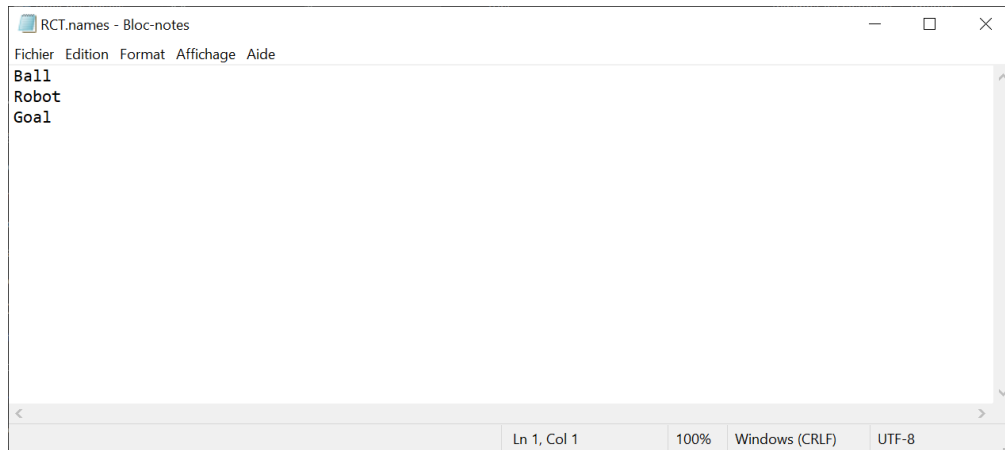
```

Fichier Edition Format Affichage Aide
RCT\train\0001.jpg
RCT\train\0003.jpg
RCT\train\0004.jpg
RCT\train\0005.jpg
RCT\train\0007.jpg
RCT\train\0008.jpg
RCT\train\0010.jpg
RCT\train\0011.jpg
RCT\train\0012.jpg
RCT\train\0013.jpg
RCT\train\0014.jpg
RCT\train\0015.jpg
RCT\train\0016.jpg
RCT\train\0019.jpg
RCT\train\0021.jpg
RCT\train\0023.jpg

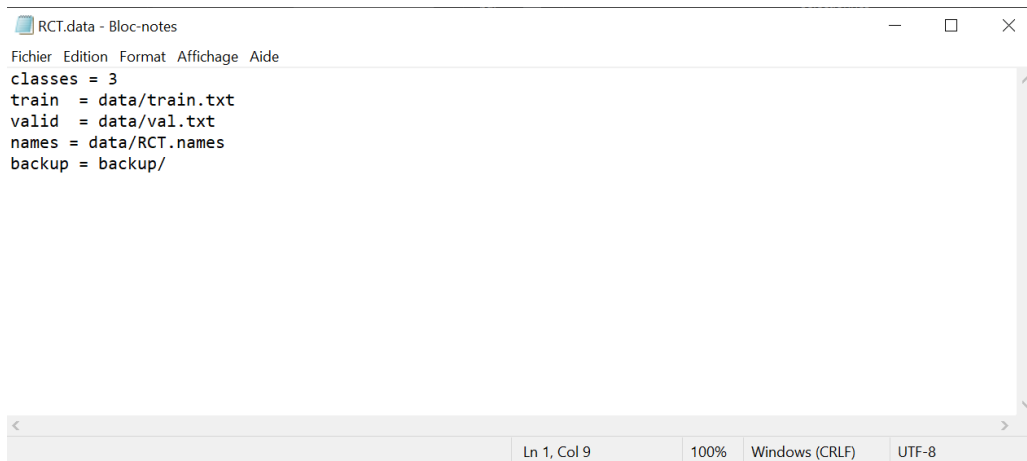
```

Ln 1, Col 1 100% Windows (CRLF) UTF-8

Créer un nouveau fichier texte, contenant le nom des différentes classes que détecte votre model. Chaque ligne correspond à une classe. Une fois toutes les classes ajouter au document texte changer son nom pour le nom de votre choix et changer son extension pour « .names ». Placer ce fichier dans le répertoire data.



Créer un nouveau fichier avec l'extension « .data » contenant les informations suivantes et organiser de la façon suivante :



Penser à changer le nombre de classes et les chemins d'accès pour qu'ils correspondent au votre. Placer le répertoire /data.

Training Yolo v4 (and v3):

Le tutorial est écrit pour les versions de YoloV4 et V3 « classique » pour les versions « tiny » certain étape son à sauter.

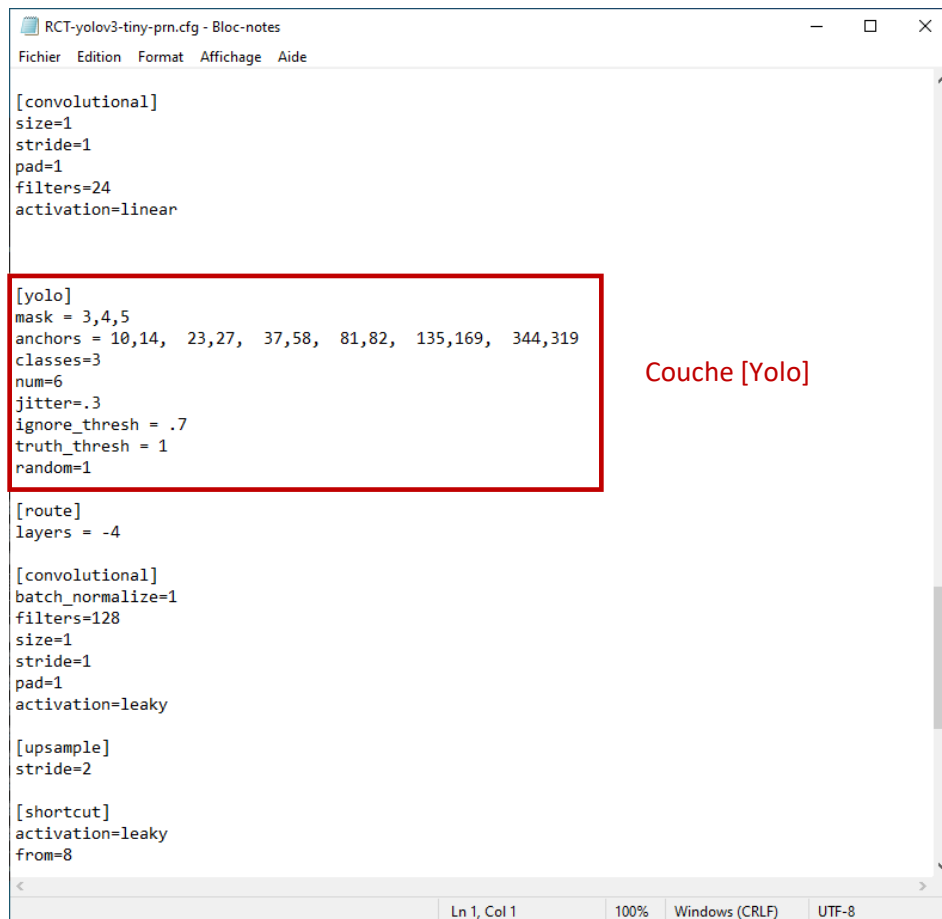
Choisir la version de Yolo et télécharger le modèle pré-entraîné :

- Modèles : [YoloV4](#)
- Modèles : [YoloV4-tiny](#)
- Modèles : [YoloV3](#)

- Modèles : [YoloV3-tiny-prn](#)
- [Autre](#)

Une fois le modèle télécharger placer le dans le même répertoire que darknet.exe. Dans le dossier /cfg faire une copie du fichier .cfg correspondant au modèle que vous avez choisie et renommer le avec le nom de votre choix. Modifier ce fichier .cfg :

- Changer la ligne « batch » à « batch=32 »
- Changer la ligne « subdivision » à « subdivision=16 »
- Changer la ligne « max_batches » à « NombreClasse * 2000 » (ne pas mettre une valeur inferieurs au nombre d'image d'entrainement et inférieur à 6000). Ex : max_batches=6000 si NombreClasse = 3.
- Changer la ligne « steps » à 80% et 90% de max_batch. Ex : steps=4800,5400 si max_batch=6000
- Changer les lignes « width » et « heigth » à « width =416 » et « heigth = 416 » ou tout autre multiple de 32.
- Changer les lignes « classes=80 » par votre nombre de classes pour les 3 couche [Yolo].



```

RCT-yolov3-tiny-prn.cfg - Bloc-notes
Fichier  Edition  Format  Affichage  Aide

[convolutional]
size=1
stride=1
pad=1
filters=24
activation=linear

[yolo]
mask = 3,4,5
anchors = 10,14, 23,27, 37,58, 81,82, 135,169, 344,319
classes=3
num=6
jitter=.3
ignore_thresh = .7
truth_thresh = 1
random=1

[route]
layers = -4

[convolutional]
batch_normalize=1
filters=128
size=1
stride=1
pad=1
activation=leaky

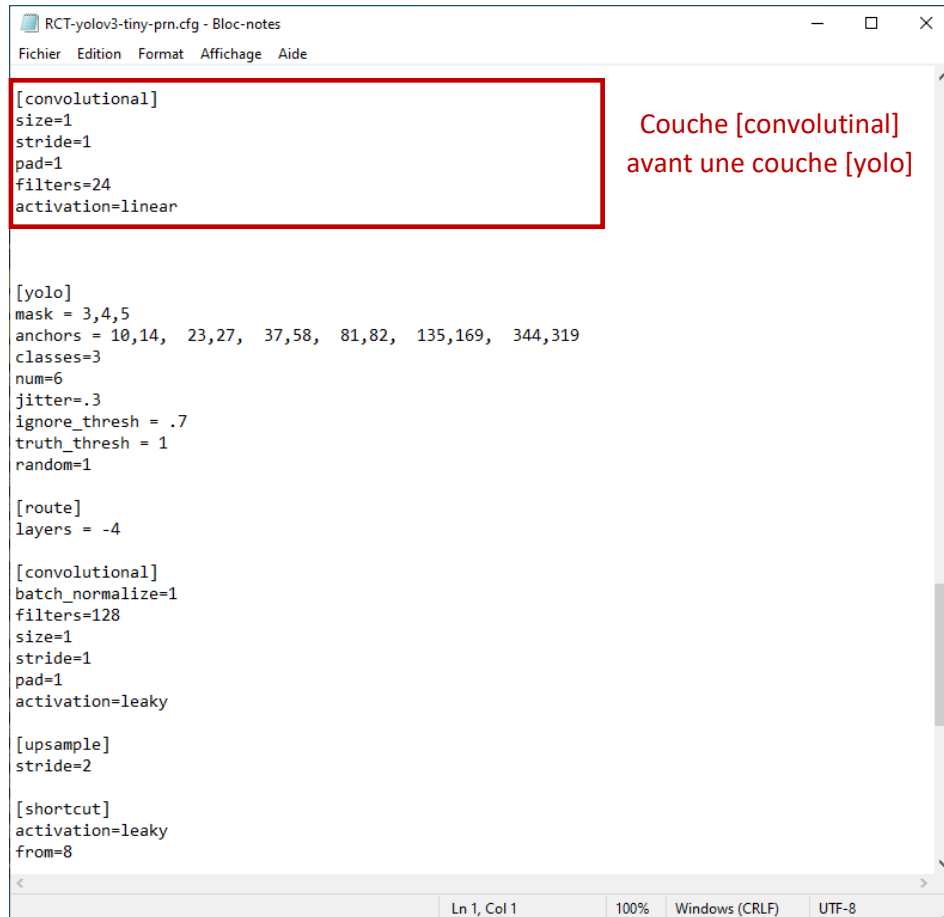
[upsample]
stride=2

[shortcut]
activation=leaky
from=8
  
```

Couche [Yolo]

Ln 1, Col 1 100% Windows (CRLF) UTF-8

- Changer les lignes « filters = 255 » pour filters = (NombreClasse + 5)*3 pour les 3 ligne bloc [convolutional] juste avant chaque couche [yolo].



```

RCT-yolov3-tiny-prn.cfg - Bloc-notes
Fichier  Edition  Format  Affichage  Aide

[convolutional]
size=1
stride=1
pad=1
filters=24
activation=linear

[yolo]
mask = 3,4,5
anchors = 10,14, 23,27, 37,58, 81,82, 135,169, 344,319
classes=3
num=6
jitter=.3
ignore_thresh = .7
truth_thresh = 1
random=1

[route]
layers = -4

[convolutional]
batch_normalize=1
filters=128
size=1
stride=1
pad=1
activation=leaky

[upsample]
stride=2

[shortcut]
activation=leaky
from=8
  
```

Couche [convolutinal]
avant une couche [yolo]

- Si votre modèle utilise des [Gaussian_yolo], changer les lignes « filters=57 » à filters=(classes + 9)x3 dans les 3 bloc [convolutional] avant chaque couche [Gaussian_yolo].

Pour lancer l'entraînement de votre modèle ouvrir l'invite de commandes et déplacer vous dans le répertoire contenant darknet.exe (Pour Windows commande « cd » suivi du chemin pour se déplacer dans le répertoire de votre choix). Puis exécuter la commande suivante :

darknet.exe detector train data/VotreNom.data cfg/VotreNom.cfg ModèleChoisie

Ex :

darknet.exe detector train data/obj.data cfg/yolo-obj.cfg yolov4.conv.137