**Etape n°1: Téléchargement et annotation des datasets**

* *Dataset PRW:*

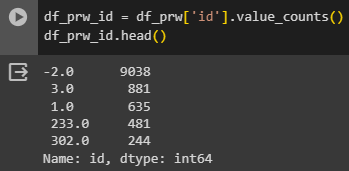
*Téléchargement du dataset:*

Importation du fichier PRW sur nos drives respectifs à partir du fichier zip fourni.

*Colonnes sélectionnées du dataset:*

Nous avons sélectionné les colonnes suivantes pour le dataframe représentant le dataset : : : img\_path, cam, frame, id, les coordonnées des bounding boxes (x1, y1, x2, y2), conf (niveau de confiance de détection), vis (seuil de visibilité).

Une particularité de ce dataset concernant les id est que "-2" signifie qu’on ne connait pas avec certitude l'identité de la personne et celui-ci n'est pas utilisé dans le test de réidentification de la personne, mais nous le conservons pour le train et test de notre modèle.



* *Dataset CAMPUS:*

*Téléchargement du dataset:*

Importation du dossier ‘multiview-object-tracking-dataset’ contenant le dataset CAMPUS depuis le lien git.

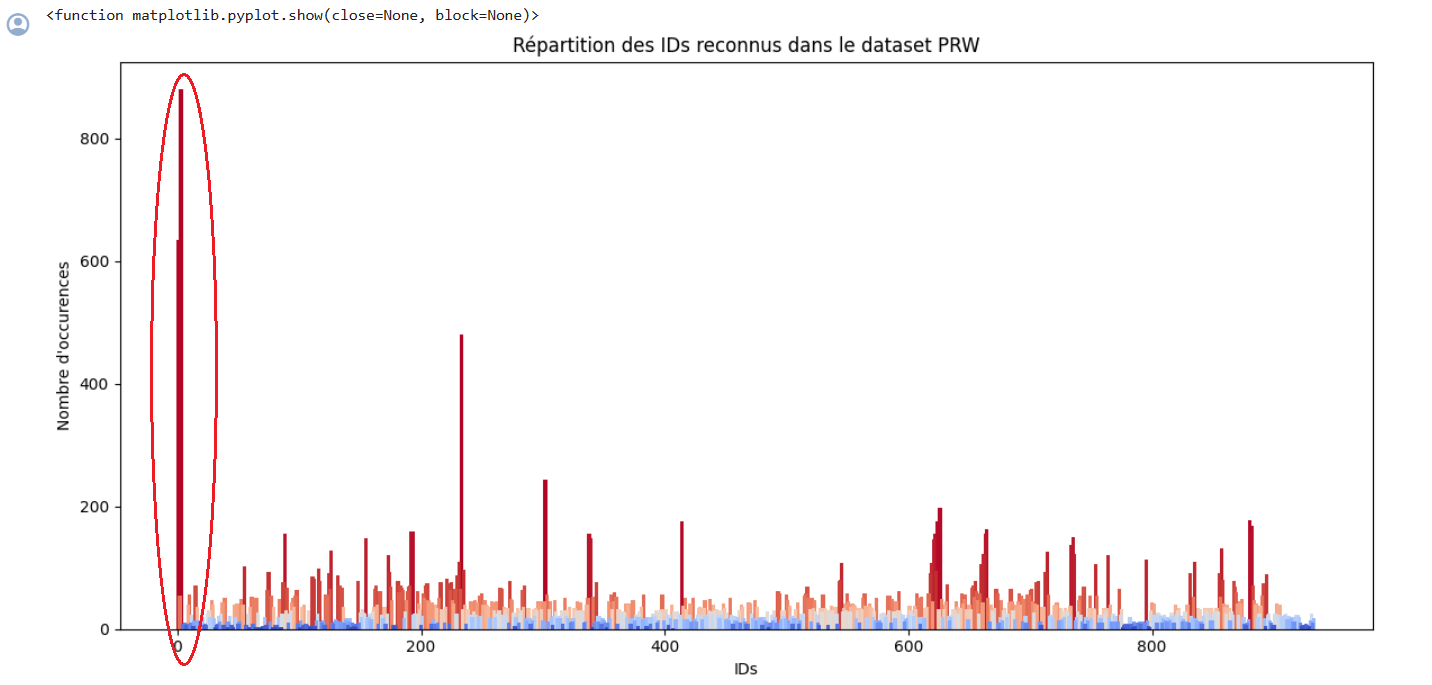
*Colonnes potentiellement sélectionnables du dataset:*

En se basant sur le Readme fourni par Yuanlu Xu, nous pouvions avoir comme informations : les coordonnées des bounding boxes (xmin, ymin, xmax, ymax) , track\_id (Id d’une personne allant de 0 à 20, mais l’ID 0 ne correspondant pas à la même personne en fonction des caméras, nous avons donc ajouté des centaines aux IDs pour chaque caméra, pour les repérer comme des chambres d’hôtel), numéro de frame, les labels des images (pour nous n’avons que des humains, donc toutes les valeurs sont ‘PERSON’), des informations sur les annotations telles que (lost, occluded, generated). Par la suite, nous avons ajouté une colonne ‘cam’ afin de différencier les différentes caméras disponibles dans le dataset.

**Etape n°2: Data visualisation**

*Histogramme : Répartition des IDs reconnus*

Analyse : Cette visualisation nous permet de voir les occurrences des IDs, afin de détecter rapidement s’il y a un très grand déséquilibre de représentations pour certains IDs et ainsi palier à un possible biais.

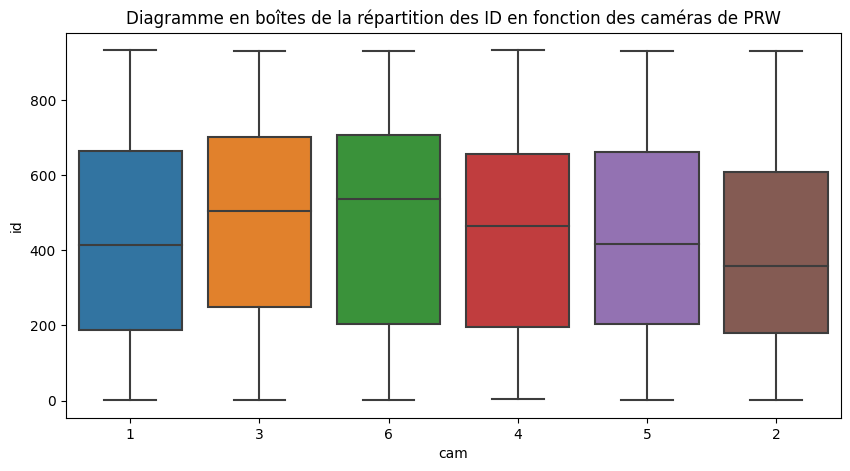


Une image contenant texte, capture d’écran, ligne, Tracé

Description générée automatiquement

*Diagramme en boîtes : Répartition des IDs en fonction des caméras*

Analyse : Même chose que le graphique ci-dessus avec l’information de la répartition des caméras cette fois-ci, et sous un format différent.



Une image contenant texte, diagramme, capture d’écran, Plan

Description générée automatiquement

*Plot : Répartition des ids en fonction de leurs coordonnées (respectivement (x1,y1) et (x2,y2) ) de leur bounding boxe*

Analyse : Ce graph était surtout un premier essai envers un graphique nous montrant l’occupation de l’espace par chaque caméra afin de déterminer si un espace n’étant jamais occupé pouvait être coupé pour réduire la complexité.

*Colormap : Répartition des bounding boxes sur chaque caméra*

Analyse : Même chose que le graph ci-dessus avec une visualisation un peu plus claire.

*Heatmap: Répartition des individus sur chaque caméra*

Analyse : Dernière version la plus appropriée pour réaliser ce que les graphes ci-dessus souhaitaient montrer. Etrangement, sur PRW, on remarque que la caméra 2 n’est jamais utilisée, on peut donc se demander pourquoi existe-t-elle dans le dataset. En dehors de cela, on remarque qu’effectivement certaines zones ne sont jamais contenues dans les box. Une approche possible pourrait etre de recouper les images de certaines caméras pour réduire le travail requis par le modèle. (A savoir que pour CAMPUS, comme nous avions des millions de données au total, nous avons arbitrairement choisi de n’étudier qu’une ligne sur 1000)

PRW

Une image contenant Caractère coloré, capture d’écran, texte, violet

Description générée automatiquement

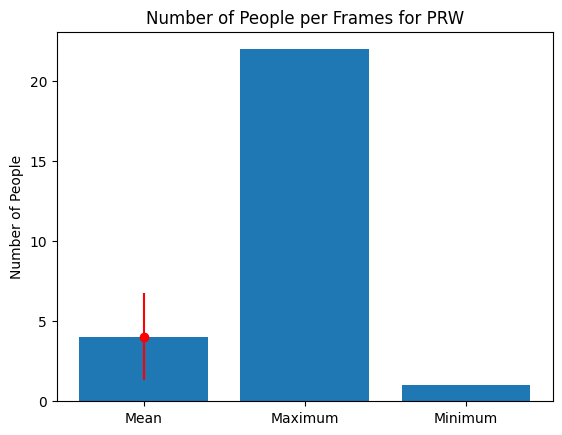
CAMPUS

Une image contenant capture d’écran, Caractère coloré, texte, violet

Description générée automatiquement

*Diagramme en baton: Nombre d’individus par cadre*

Analyse : Cette visualisation permet de voir la répartition des individus par frame en ayant comme information le nombre d’individus maximum et minimum.



Une image contenant texte, capture d’écran, diagramme, Tracé

Description générée automatiquement

*Image du dataset : Affichage aléatoire d’une des images téléchargées avec les bounding box*

Analyse : On peut voir à travers cette visualisation que nos images ont été correctement importées et intégrées dans le dataframe créé.

PRW

Une image contenant plein air, capture d’écran, ciel, personne

Description générée automatiquement

CAMPUS

Une image contenant arbre, plein air, habits, personne

Description générée automatiquement

**Etape n°3: Création des structures pytorch**

Nous n’avons réalisé cette partie qu’avec le dataset PRW, car nous avons eu du mal à comprendre certaines subtilités du dataset CAMPUS.

Nous avons commencé par créer notre classe Dataset héritant de la classe éponyme de PyTorch, en adaptant la méthode \_\_getitem\_\_ et en fabriquant notre propre transform.Compose afin de pouvoir plus facilement fabriquer notre batch d’images ayant toutes la même taille, et étant déjà centrée sur la bounding box indiquée par les coordonnées fournies dans le dataset.

Nous chargeons ensuite les csv de notre dataset dans la nouvelle classe Dataset, en ayant au préalable séparé notre dataset en 2 parties de manière aléatoire, une pour le train, et une pour le test. Le dataset train fait 0.7 fois la taille du dataset originel.

On finit par créer des indices pour le training et la validation, les data samplers et loaders, puis pour vérifier que tout s’est bien implémenté, on affiche un batch au hasard avec les IDs que notre modèle a pu identifier.

PRW

Une image contenant collage, texte, habits, personne

Description générée automatiquement

CAMPUS

Une image contenant habits, texte, collage, personne

Description générée automatiquement

Bibliographie

Xu, Y., Liu, X., Liu, Y., & Zhu, S. C. (2016). Multi-view people tracking via hierarchical trajectory composition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4256-4265).[En ligne].[Consulté le 03/12/2023] Disponible à l'adresse : <https://bitbucket.org/merayxu/multiview-object-tracking-dataset/src/master/>

Xu, Y., Liu, X., Qin, L., & Zhu, S. C. (2017, February). Cross-view people tracking by scene-centered spatio-temporal parsing. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 31, No. 1).[En ligne].[Consulté le 03/12/2023] Disponible à l'adresse : <https://bitbucket.org/merayxu/multiview-object-tracking-dataset/src/master/>

Zheng, L., Zhang, H., Sun, S., Chandraker, M., Yang, Y., & Tian, Q. (2017). Person re-identification in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1367-1376).[En ligne].[Consulté le 03/12/2023] Disponible à l'adresse : <http://zheng-lab.cecs.anu.edu.au/Project/project_prw.html>