

Photomosaïque avec critères avancés

Présentation du projet

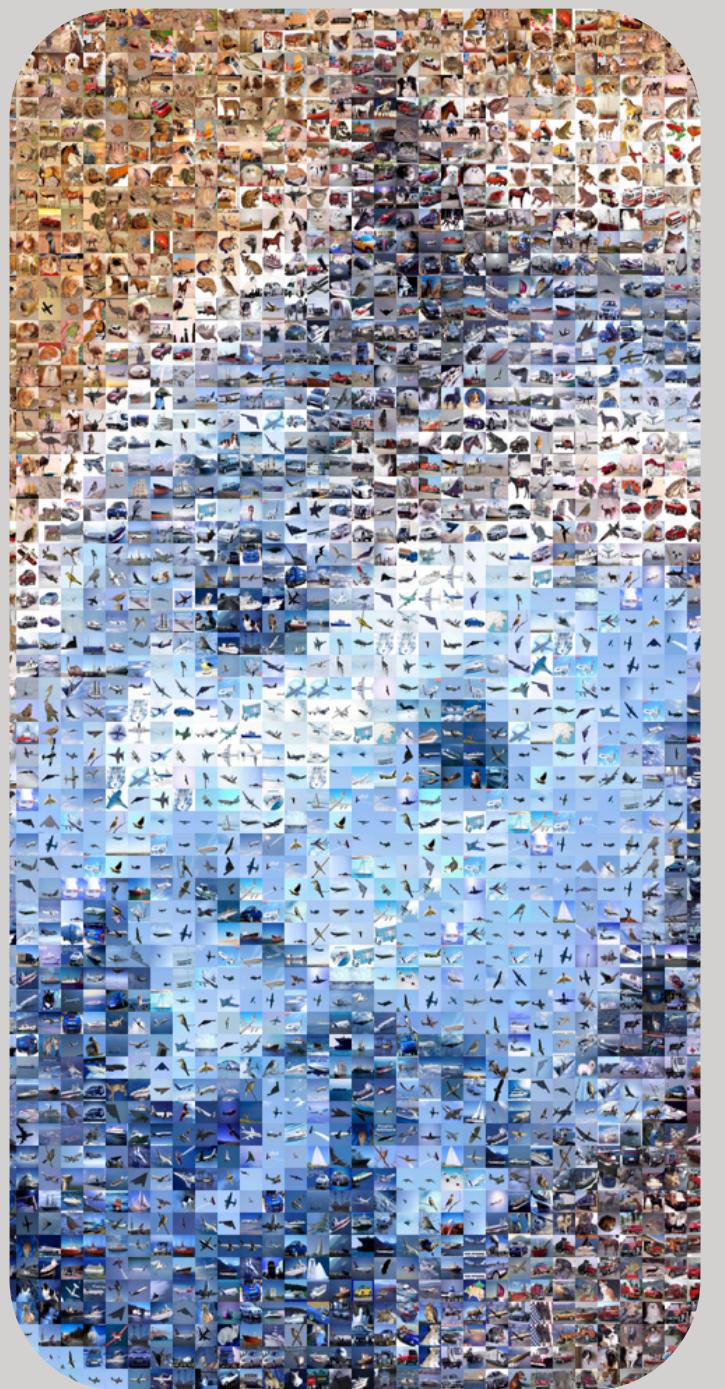
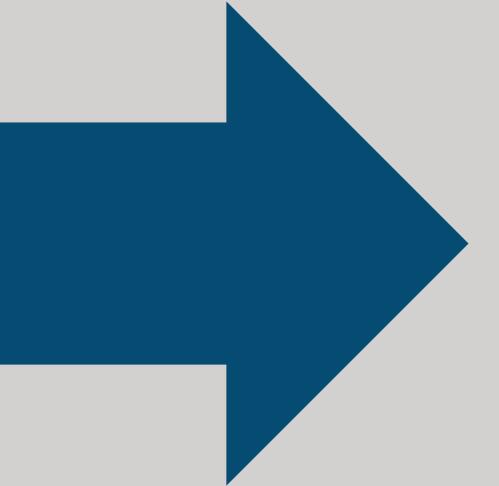
CANHOTO Mickaël, BELDJILALI Maxime

Sommaire

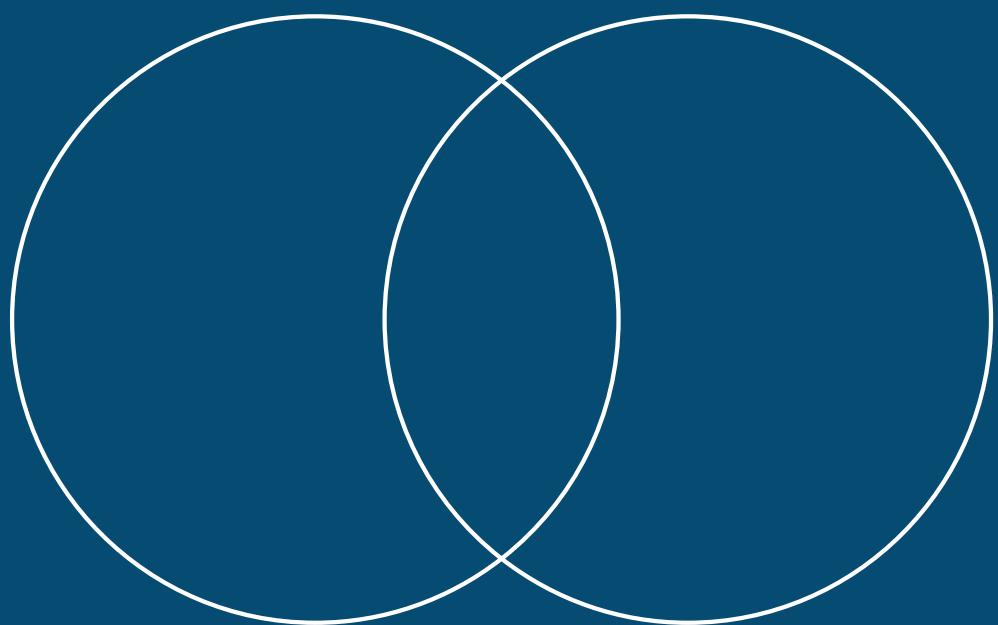
- **Introduction**
- **Traitemet du Dataset**
- **KDTree**
- **Correspondance sans répétitions**
- **Gestion des vidéos**
- **Interface graphique**
- **Pistes d'améliorations**
- **Conclusion**

Introduction

La photomosaïque est une photo reconstituée par la juxtaposition, à la manière d'une mosaïque, d'un grand nombre d'images beaucoup plus petites.



Méthode



4 étapes générales:

1. Collecte d'attributs sur une large collection d'images
2. Segmentation en blocs de l'image
3. Correspondance entre les attributs des blocs et ceux de la collection d'images
4. Reconstruction de l'image

Programme : Mosaique-images

Disponible sur github : <https://github.com/Pataleon/Mosaique-images>

Développer sur python avec les librairies :

- Numpy
- OpenCv
- Pillow
- Scipy



Choix du
dataset ?

Choix 1: Collection d'images simple (ex : div2k)

- Plus facile à modifier
- Permet de choisir ses images
- Nécessite de redimensionner chaque images

Choix 2: Collection d'images de petites tailles uniformes (Cifar 10)

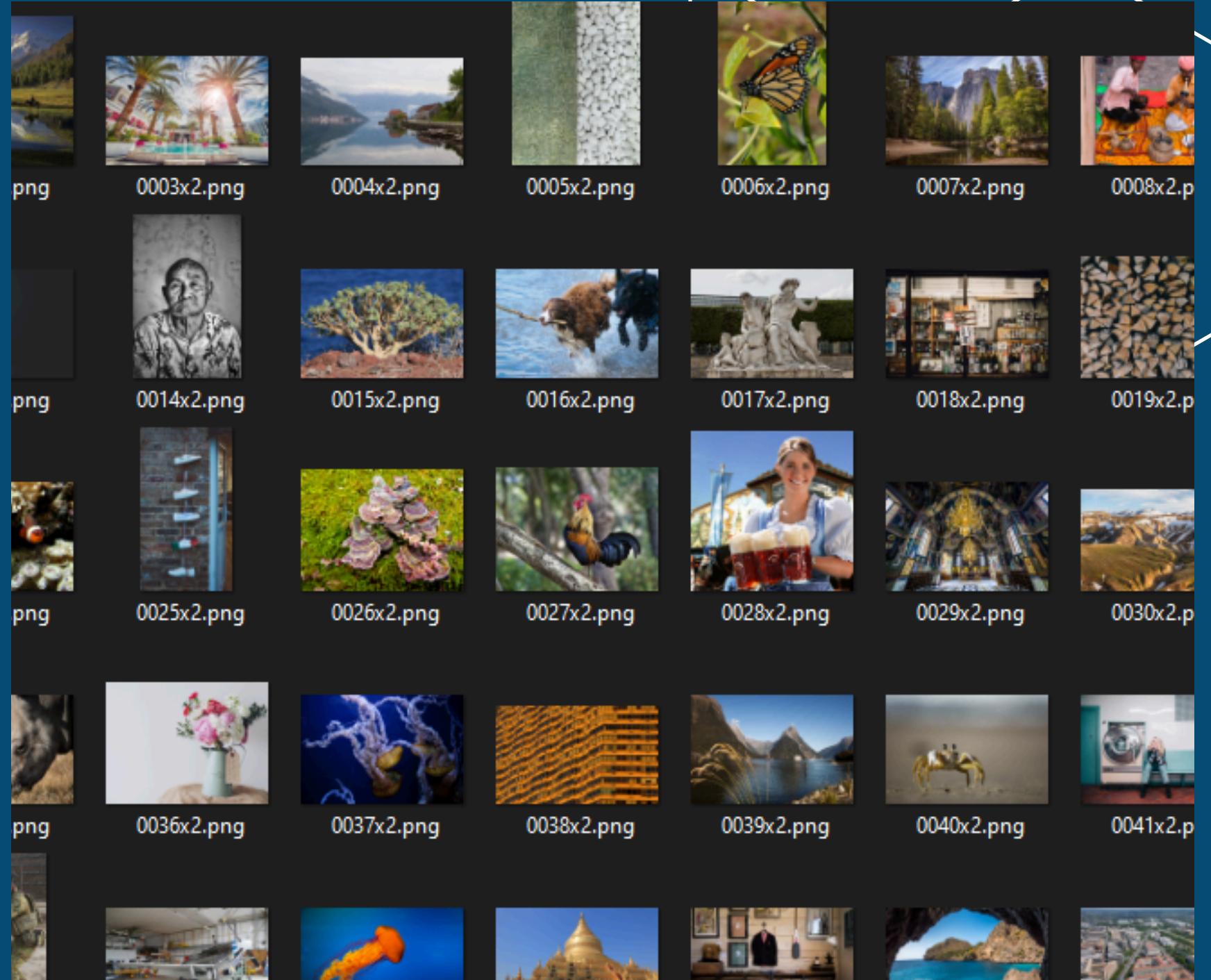
- 60 000 images de tailles 32x32 donc pas de redimensionnement
- Directement écrit dans des arrays numpy donc plus rapide
- La phase de prétraitement du dataset avec la moyenne comme critère gagne en vitesse d'exécution. ~15s contre ~0.7s

Traitement du dataset

Si collection d'images à tailles variables,
redimensionnement

Pour chaque images, calculer la moyenne
(ou autres critères)

Construire une KD Tree avec cet ensemble
d'attributs



KD Tree

Pour le critère établie :

01

Création d'un arbre de recherche

03

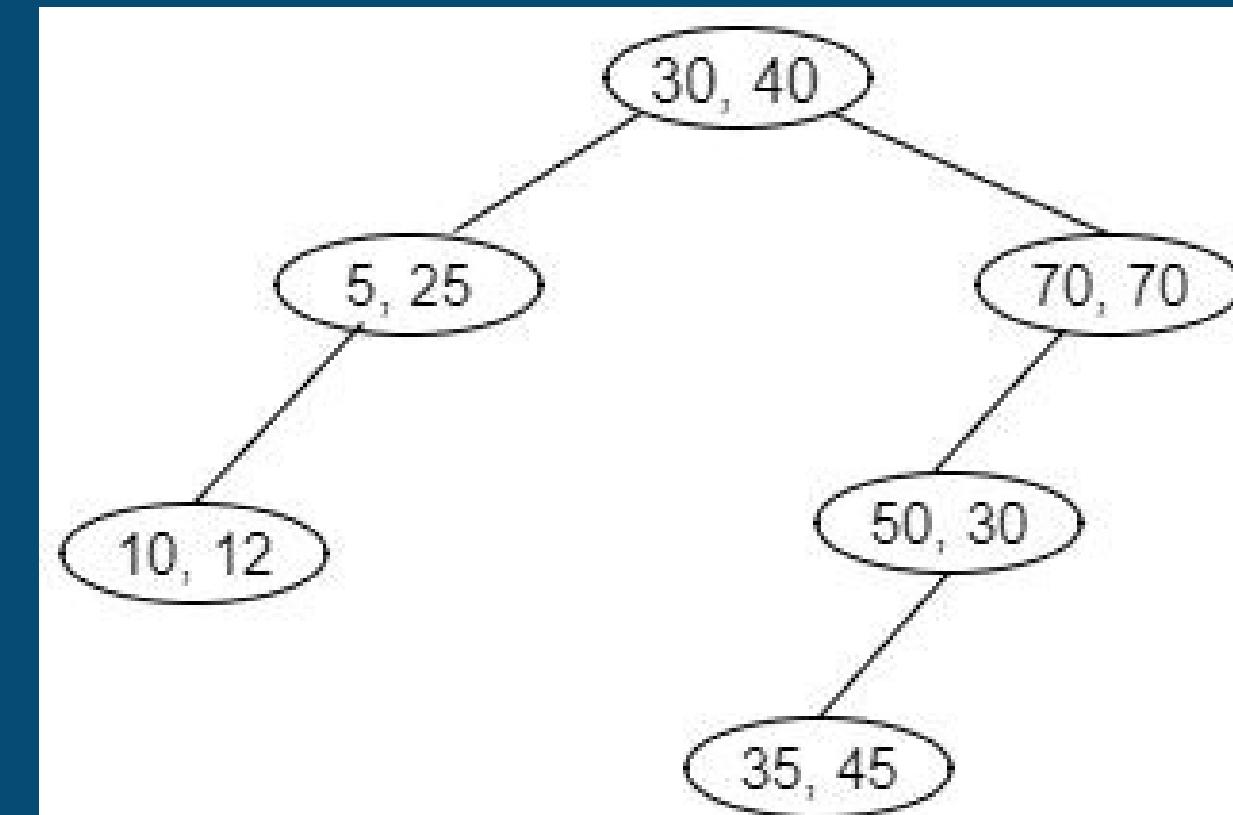
On applique le/un des meilleurs choix possibles.

02

Pour chaque mosaïque, on fait un match avec l'arbre

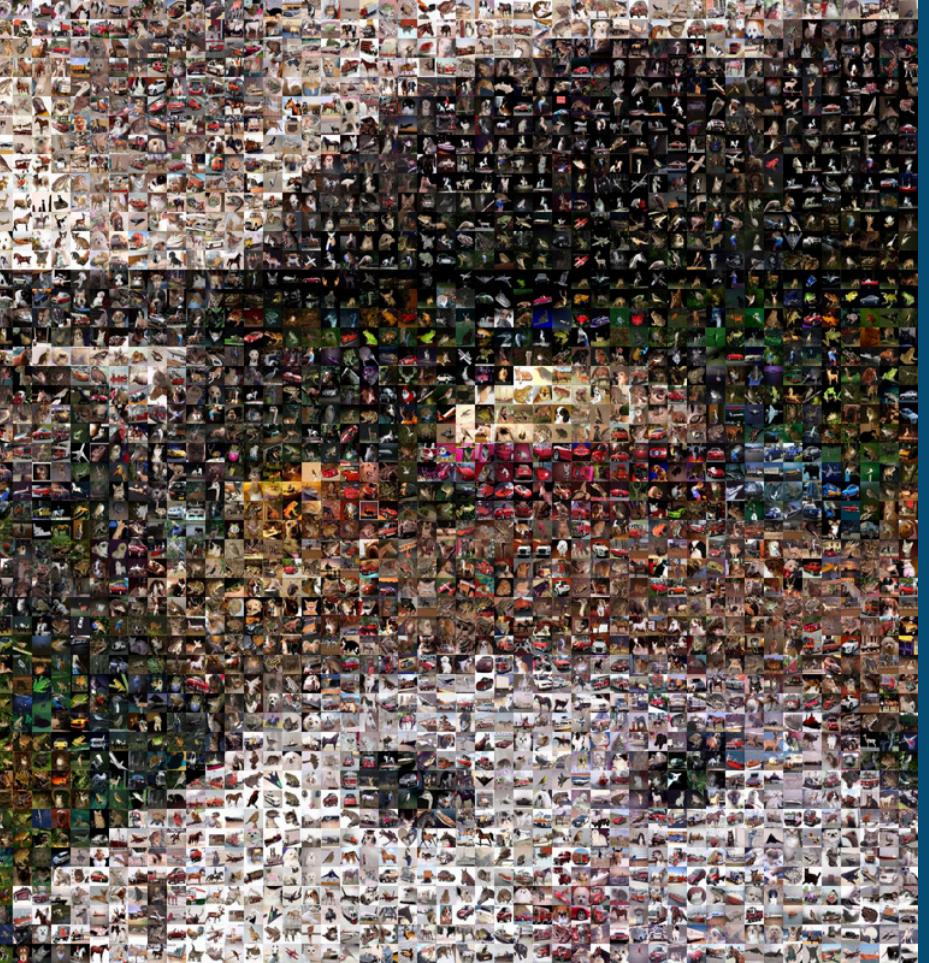
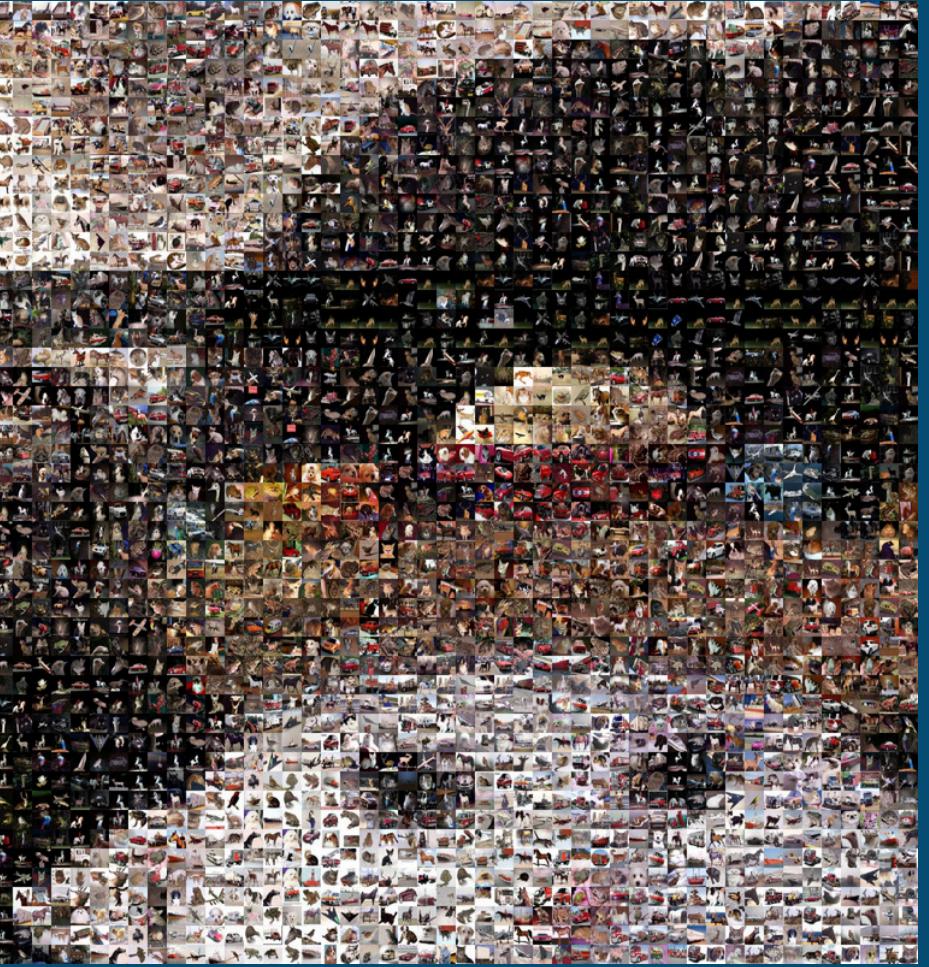
04

Parcours un arbre binaire au lieu d'une liste.



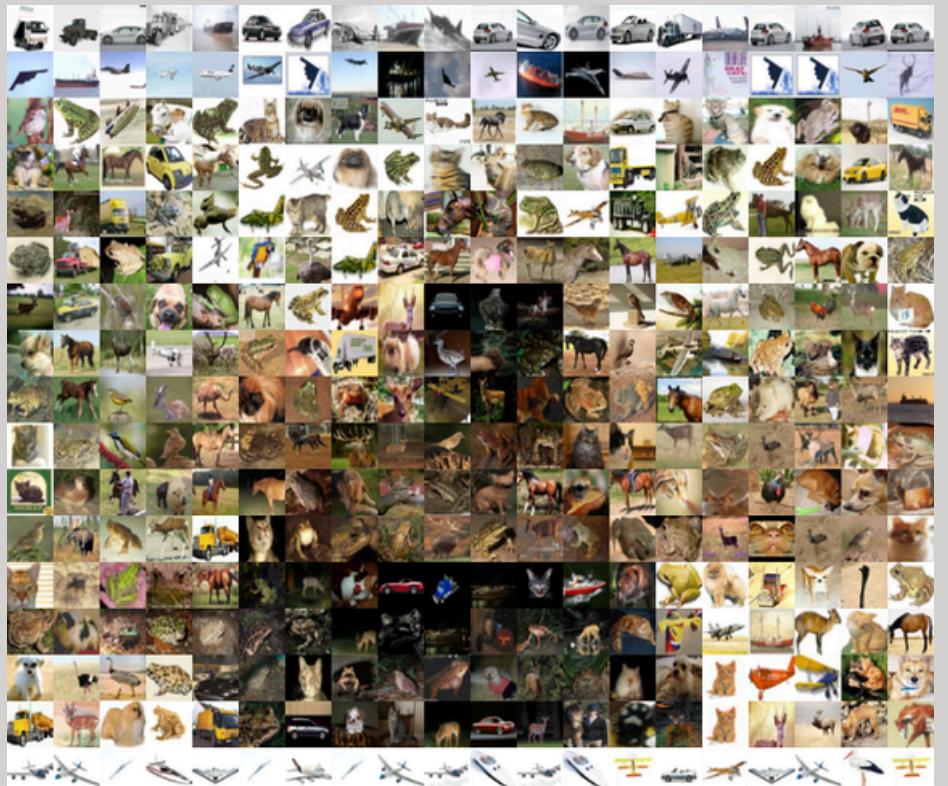
Correspondance sans répétitions

- Changement d'algorithme de correspondance pour une version lente, mais qui évite les répétitions d'images dans la photo mosaïque.

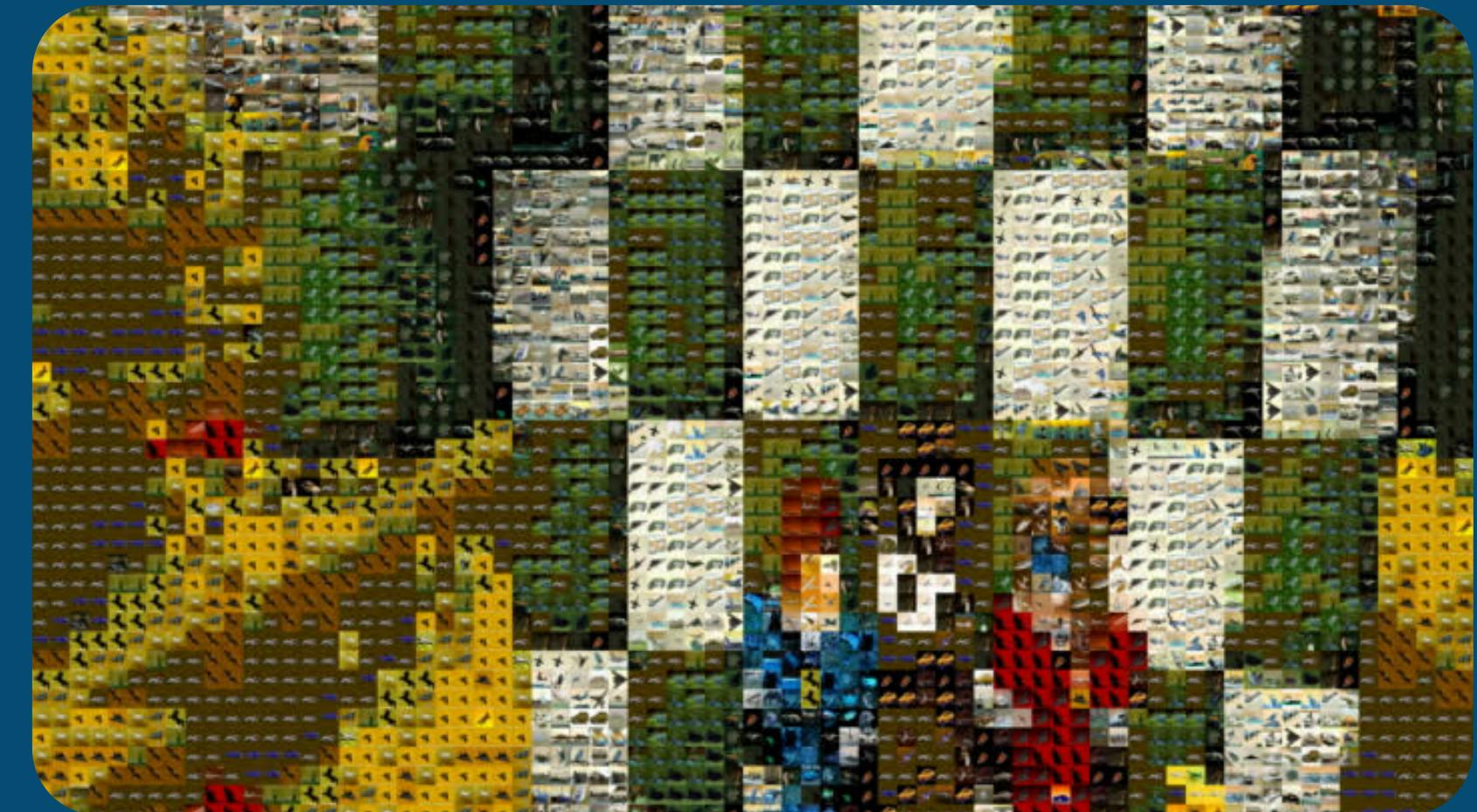
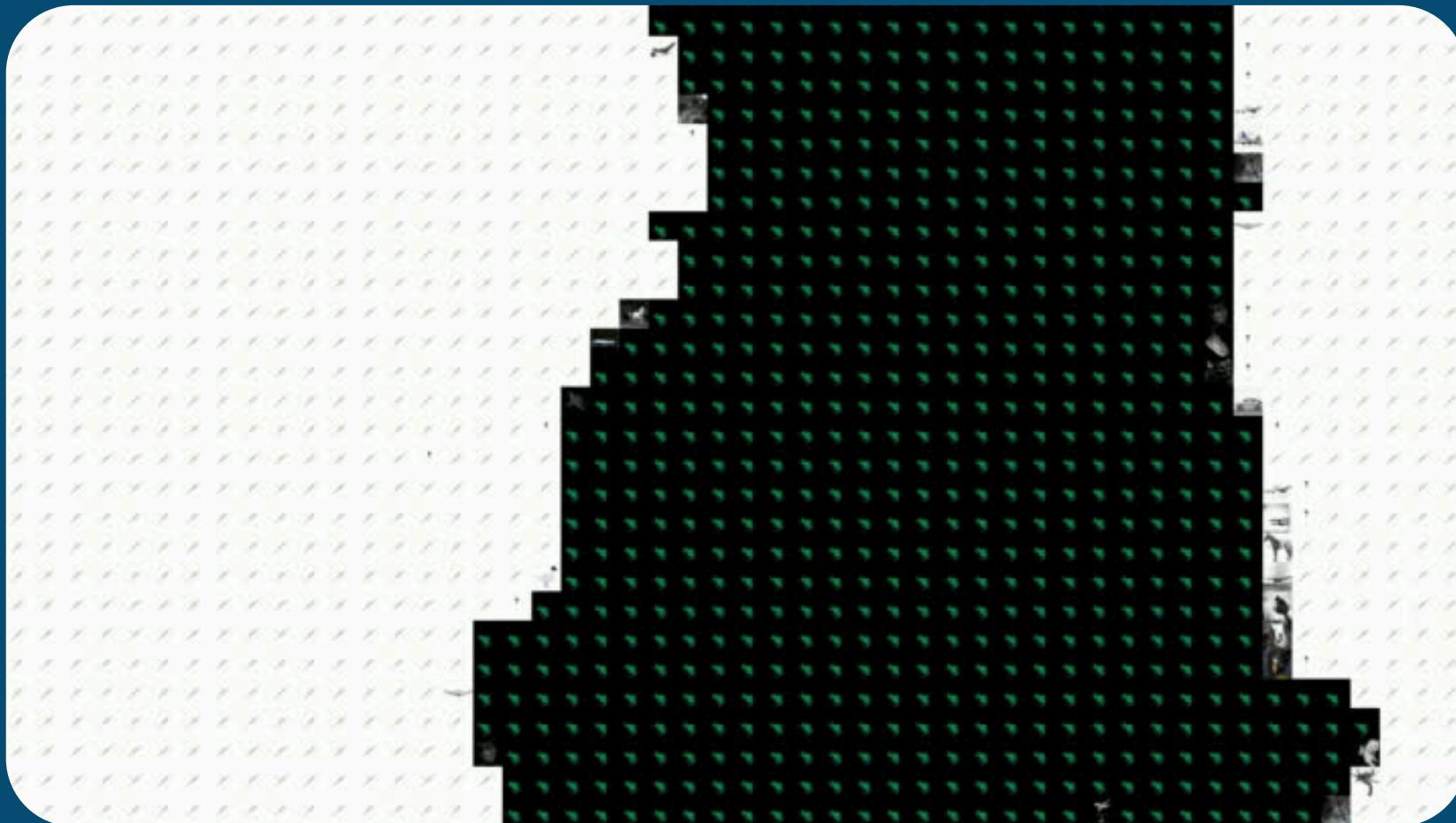


Qualité du résultat/ Temps de traitement

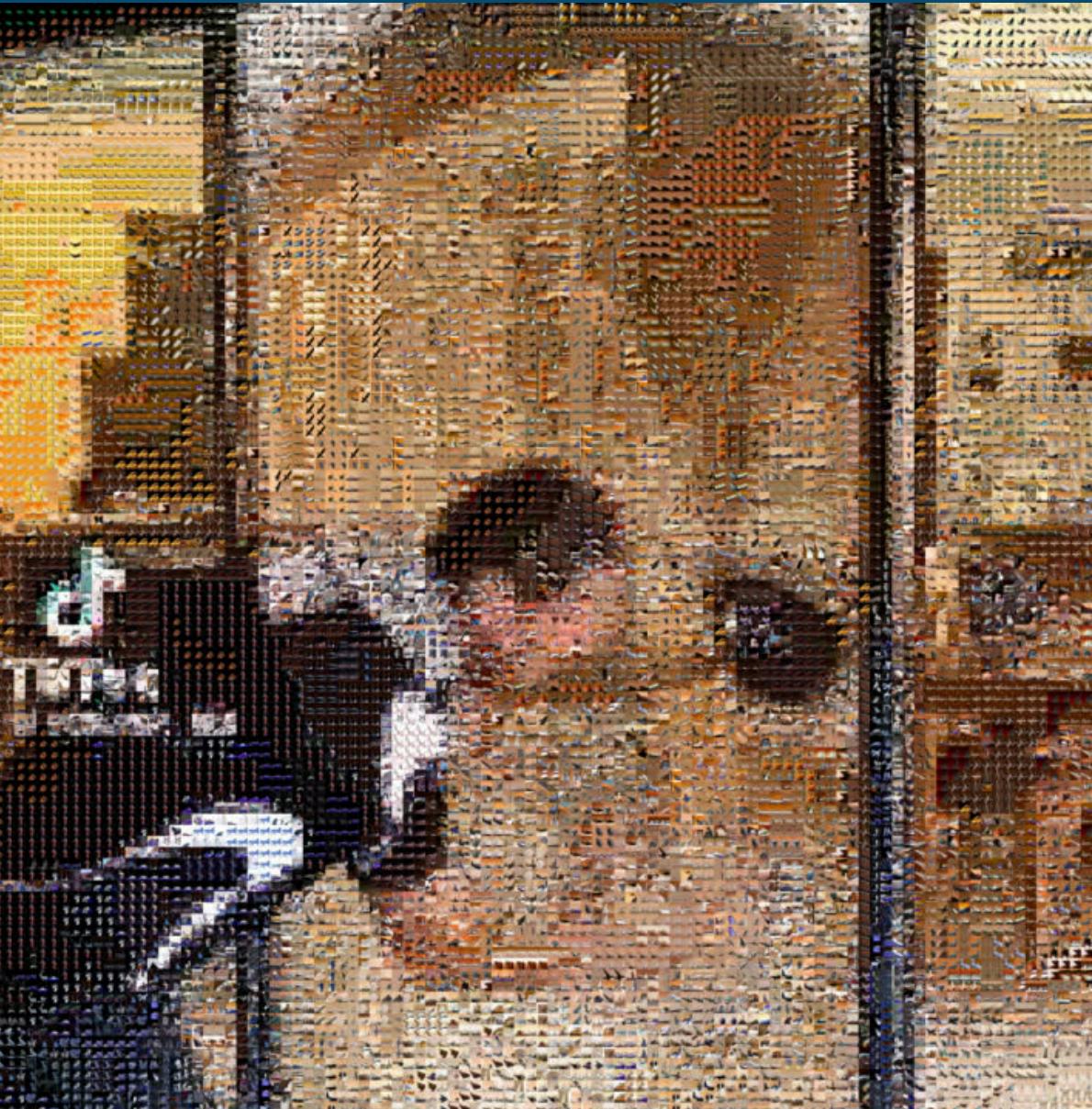
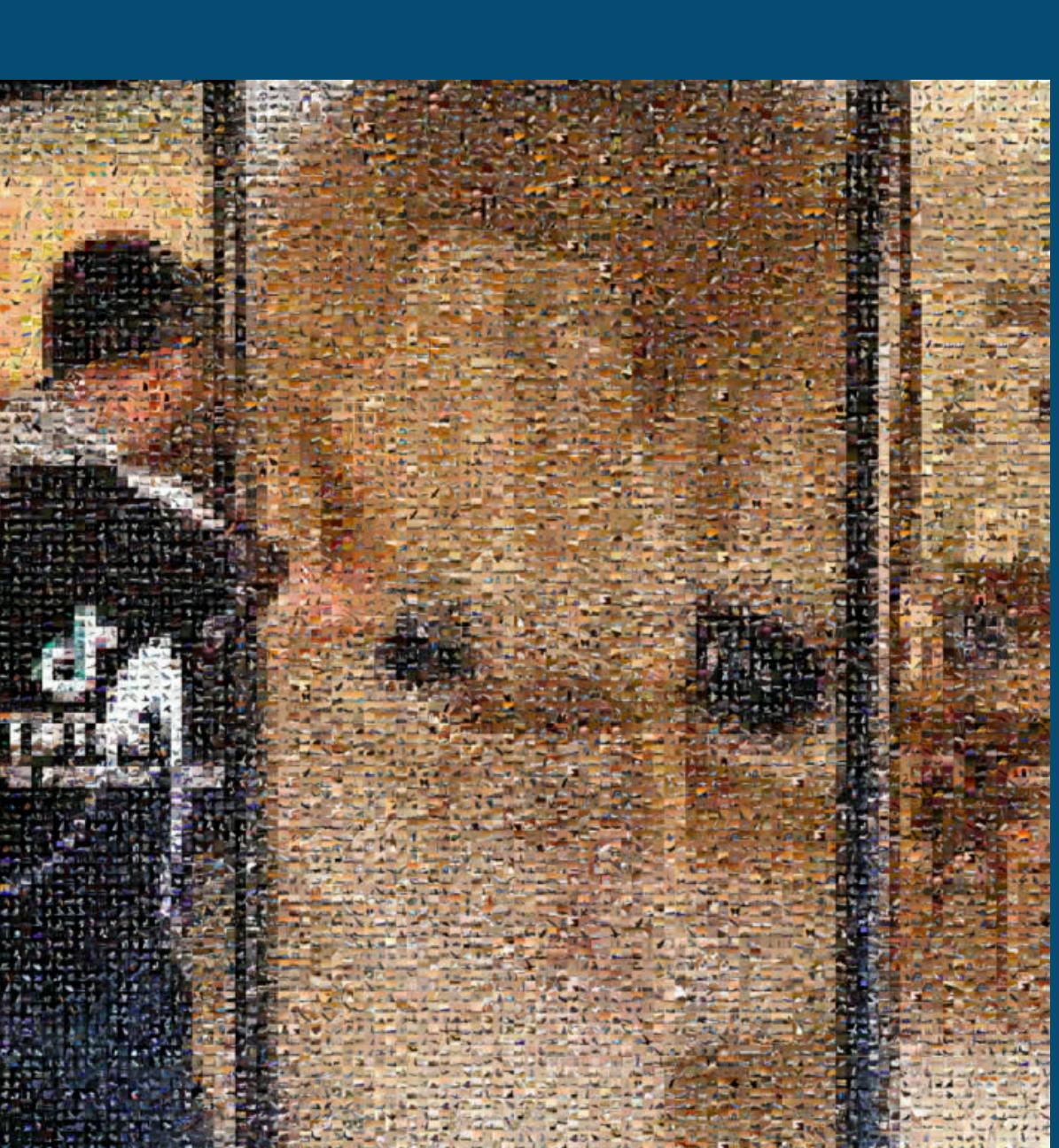
- Résultat visuel moins sans répétitions
- Temps de traitement plus long car complexité temporelle moins bonne
- Choix entre qualité visuelle/compression



Gestion d'une vidéo

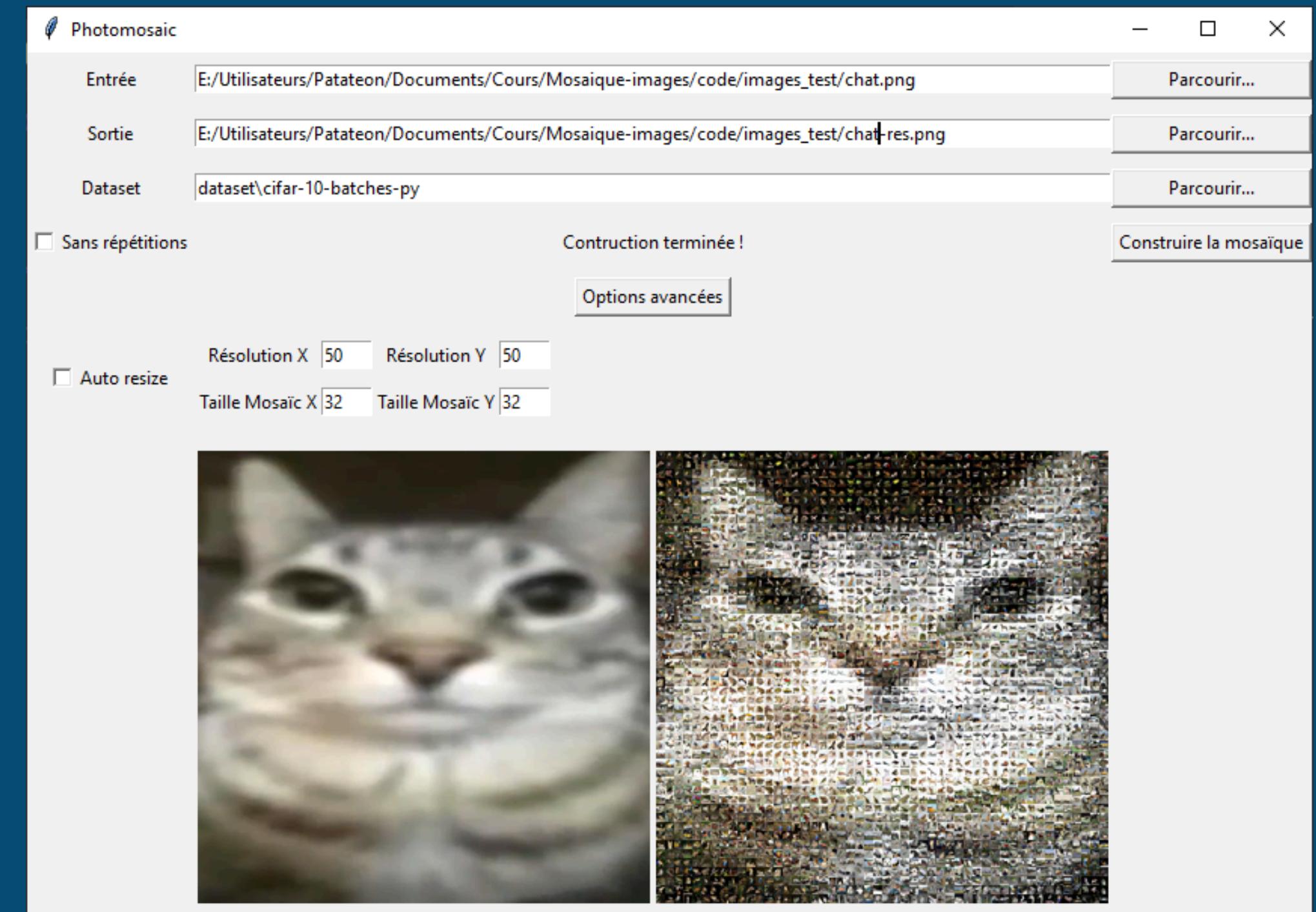


Nouveau format, nouvelles difficultés

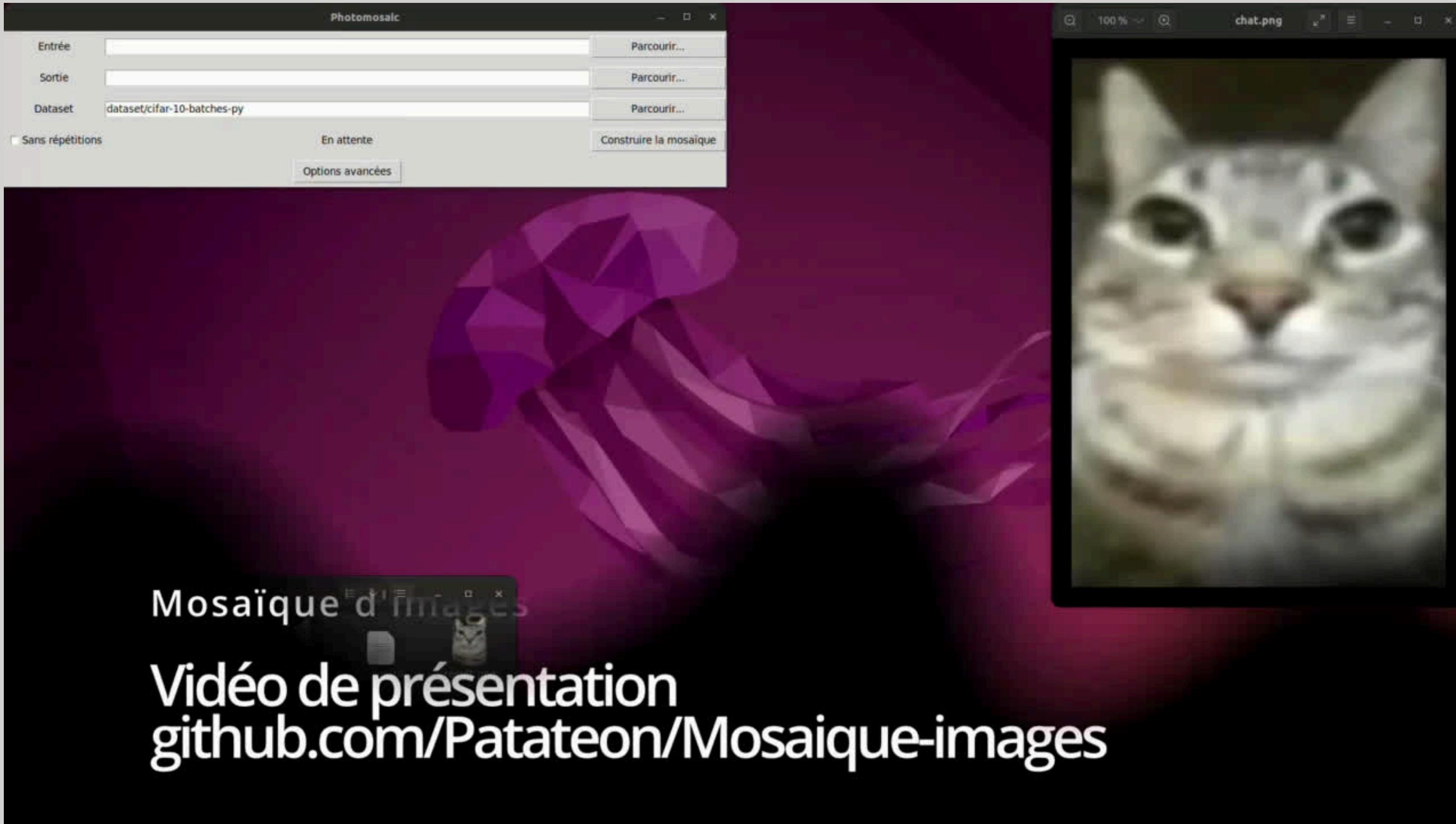


Implémentation d'une interface graphique

- Simplifier l'utilisation
- Permettre de choisir un dataset
- Alterner avec une méthode de matching



Démonstration





Pistes d'améliorations

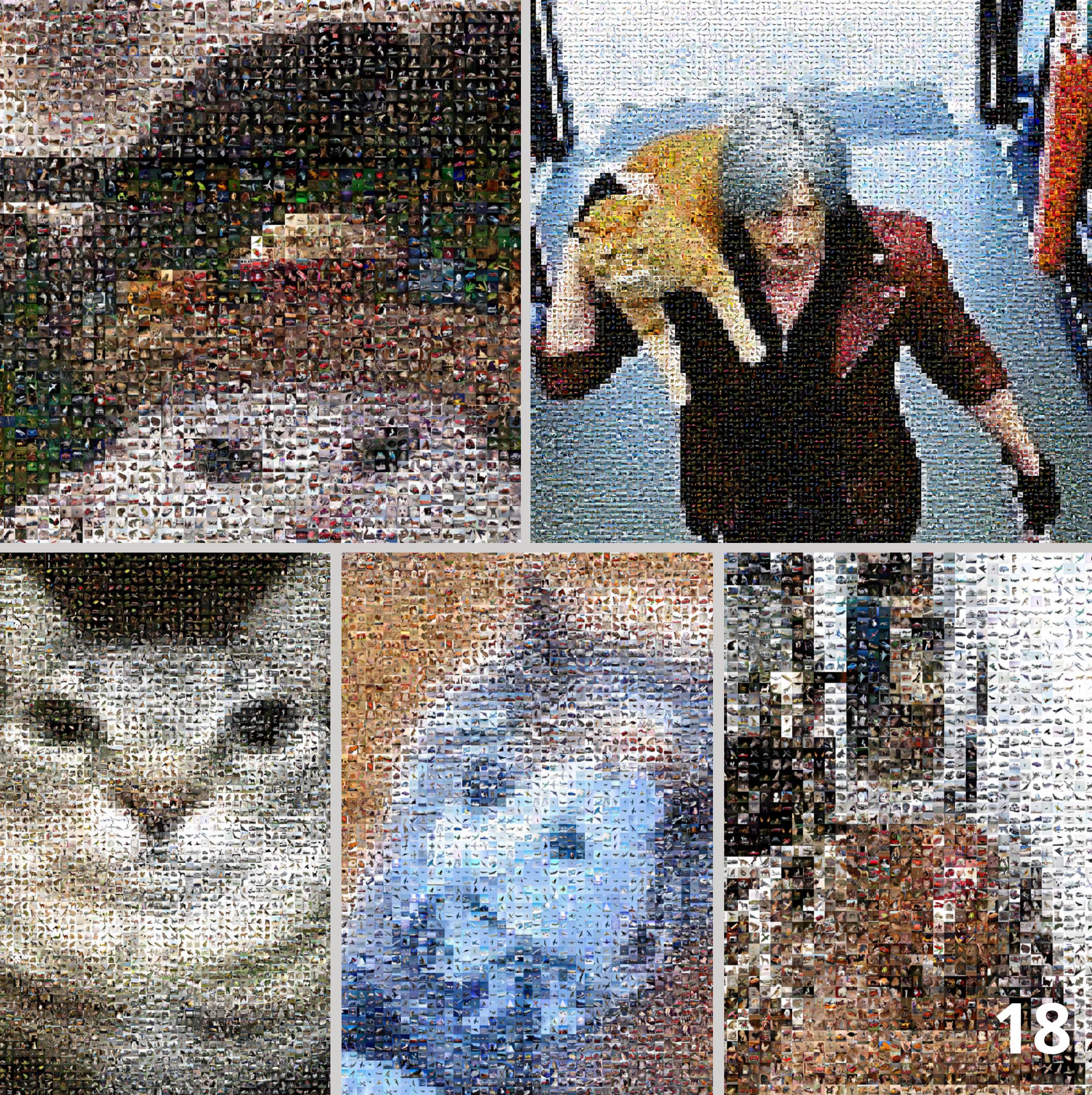
Meilleur algorithme de correspondance

- Composing photomosaic images using clustering
- Randomized Iterative Algorithm

Prédiction spatiale

Accélérer la génération d'une vidéo à l'aide de prédiction spatiale

Conclusion



Références

- [1] - Blasi, Gianpiero & Petralia, Maria. (2005). Fast Photomosaic :
<https://github.com/Patareon/Mosaique-images/blob/main/documents/Fast%20Photomosaic.pdf>
- [2] - Di Blasi, G., Gallo, G. Artificial mosaics. Visual Comput 21, 373–383 (2005) :
<https://github.com/Patareon/Mosaique-images/blob/main/documents/Artificial%20mosaics.pdf>
- [3] - Kim J. and Pellacini F. Jigsaw Image Mosaics. In proceedings of SIGGRAPH2002, pp 657-664, 2002 : <https://www.cs.cornell.edu/people/jkim/JIM/JigsawImageMosaics.pdf>
- [4] - Battiatto, Sebastiano & Gianpiero, Di & Blasi, Giovanni & Maria, & Farinella, Giovanni & Gallo,. Digital Mosaic Frameworks -an overview (preprint version) :
https://www.researchgate.net/publication/266405610_Digital_Mosaic_Frameworks_-an_overview_preprint_version
- [5] - H. Narasimhan and S. Satheesh, "A randomized iterative improvement algorithm for photomosaic generation," 2009 World Congress on Nature & Biologically Inspired Computing (NaBIC), Coimbatore, India, 2009, pp. 777-781, doi: 10.1109/NABIC.2009.5393882.
- [6] - He, Y., Zhou, J. & Yuen, S.Y. Composing photomosaic images using clustering based evolutionary programming. Multimed Tools Appl 78, 25919–25936 (2019).
<https://doi.org/10.1007/s11042-019-07798-5>
- [7] - Y. He, J. Zhou, and S. Y. Yuen, "Composing photomosaic images using clustering based evolutionary programming," Multimedia Tools and Applications, vol. 78, no. 18, pp. 25919–25936, 2019.
- [8] - Narasimhan H, Satheesh S (2009) A randomized iterative improvement algorithm for photomosaic generation. In: Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on, IEEE, pp 777–781
- [9] - Lee HY (2017) Automatic photomosaic algorithm through adaptive tiling and block matching. Multimedia Tools and Applications 76(22):24281– 24297

