

# Rapport- Entretien technique – Ingénieur d'étude

## Data / préparation

### Partie 1

L'article portait sur *VideoSum*, une librairie python utilisée pour « résumer » des vidéos sous forme de storyboards. Ici, un storyboard est un collage constitué d'un échantillon d'images clés (frame) extraites de la vidéo et représentatif de celle-ci.

En effet, d'après l'article, le type de vidéo utilisée correspond à des enregistrements recueillis lors d'opérations chirurgicales en bloc opératoire. Ces dernières sont généralement longues avec une durée moyenne d'environ 130.45 minutes. Ainsi, le nombre de frames contenus dans une vidéo peut être très important, d'où la nécessité de « résumer » celle-ci au moyen de frame clé tout en conservant l'essence de la donnée.

Il est donc important d'avoir des images claires, bien éclairées et sans artefact majeurs. Bien que la résolution n'ait pas été fixée à une valeur précise, il convient tout de même de s'assurer que celle-ci soit suffisante pour le bon fonctionnement des méthodes d'extraction de features de *Videosum*.

Bien évidemment, dans une optique d'analyse des mouvements du chirurgien et de son expertise, il convient de s'assurer que les certains éléments chirurgicaux clés (instruments, champ opératoire, etc.) sont bien apparents sur les vidéos.

Pour pouvoir tester les algorithmes à partir des vidéos de vos vidéos de bloc opératoire, il faut s'acquitter d'un certain nombre condition.

Il faudra notamment s'assurer que celles soient échantillonnées à une frame par seconde (1 fps), à l'image des vidéos utilisées dans l'article. La caméra doit être également stabilisée.

*Videosum* requiert un environnement technique précis. En effet, pour pouvoir fonctionner correctement, il nécessite non seulement un GPU mais aussi l'installation d'un certain nombre de dépendance :

- Ubuntu  $\geq 20.04$
- Python  $\geq 3.10$  supportant NumPy-MKL
- Pytorch 1.12.1 +cu116
- ffmpeg
- swig
- faiss-gpu : son installation nécessite l'installation des sous dépendances suivantes :
  - CMake
  - Compilateur C++17
  - Une Implémentation BLAS
  - CUDA toolkit

Ces dépendances peuvent être installer directement en local ou via un conteneur Docker.

En outre, l'utilisateur devra fournir le chemin vers la vidéo ainsi que celui où le storyboard sera enregistré. Il est également possible de faire varier certains paramètres tels que :

- le nombre de frame que vous voudriez voir dans le collage final,
- les dimensions du collage (hauteur, largeur)
- la fréquence d'échantillonnage requise pour sous échantillonner la vidéo initiale
- la méthode à utiliser pour la sélection des frames clés. 4 méthodes d'apprentissage non supervisées sont proposées : *time*, *inception*, *uid*, *scda*. La méthode recommandée, produisant un storyboard le plus présentatif de la vidéo en un temps d'exécution optimal est *inception*.

## Partie 2

Il m' été malheureusement impossible de tester les solutions proposées dans cette partie, sur mon ordinateur personnel .

En effet, un environnement Linux était nécessaire, sachant que mon matériel est sous Windows.

Bien que mon ordinateur soit doté d'un GPU et permet en théorie la virtualisation, celui-ci a été contraint d'être redémarrer à plusieurs reprises, en mode sans échec, lors de l'installation d'un conteneur Docker.

J'ai également essayé d'installer les dépendances localement, mais la dernière ((faiss-gpu)) nécessitait l'activation de la virtualisation. Cela me ramenait donc au problème cité précédemment lors de l'installation du conteneur Docker.

## Méthodologie d'annotation semi-automatique des pinces chirurgicales et comparaison des outils (MedSam, CVAT).

Afin de pouvoir comparer les outils proposés, il est nécessaire d'avoir un dataset annoté manuellement dit *ground truth*. Ainsi, une fois le storyboard résumant la vidéo à disposition, les frames clés pourront être annotées manuellement par un expert. La *ground truth* sera nécessaire en tant que référence pour l'évaluation des performances.

En effet, CVAT est une plateforme permettant de réaliser des annotations manuelles ou semi-automatiques d'images. Ainsi, CVAT pourra être utilisé pour obtenir la groundtruth. Mais également pour générer un dataset de frames annotées semi-automatiquement.

Quant à MedSam, il s'agit d'un modèle de segmentation adapté pour l'imagerie médicale. Dans ce cas, MedSam, pourra être utilisé pour obtenir un masque de segmentation précis des pinces.

L'évaluation peut se faire sur la base de critères tel que : la précision de l'annotation, le temps d'annotation global, la quantification de l'effort de correction humaine, coefficient de Dice, IoU.

## Data / Synchronisation

Cette partie a été traitée principalement dans le Jupyter Notebook associé à ce rapport.