La compression vidéo et l'IA: réalité et fiction

Luc Trudeau





Luc Trudeau

Chercheur en compression vidéo

Coauteur du format (AVA)



- Auteur de brevets et d'articles scientifiques



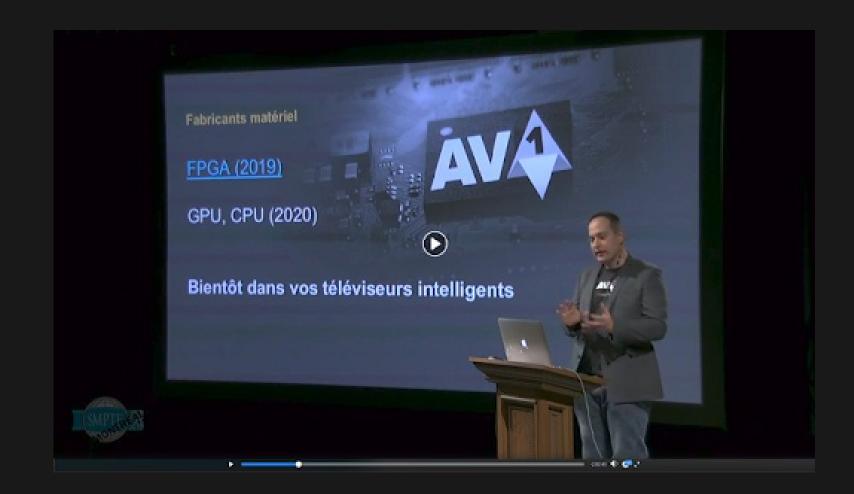
• Coconcepteur du décodeur 🕝 💄





• Gradué de l'ÉTS (bac., maîtrise, PhD.)







La plus grande contribution de l'IA à l'industrie?

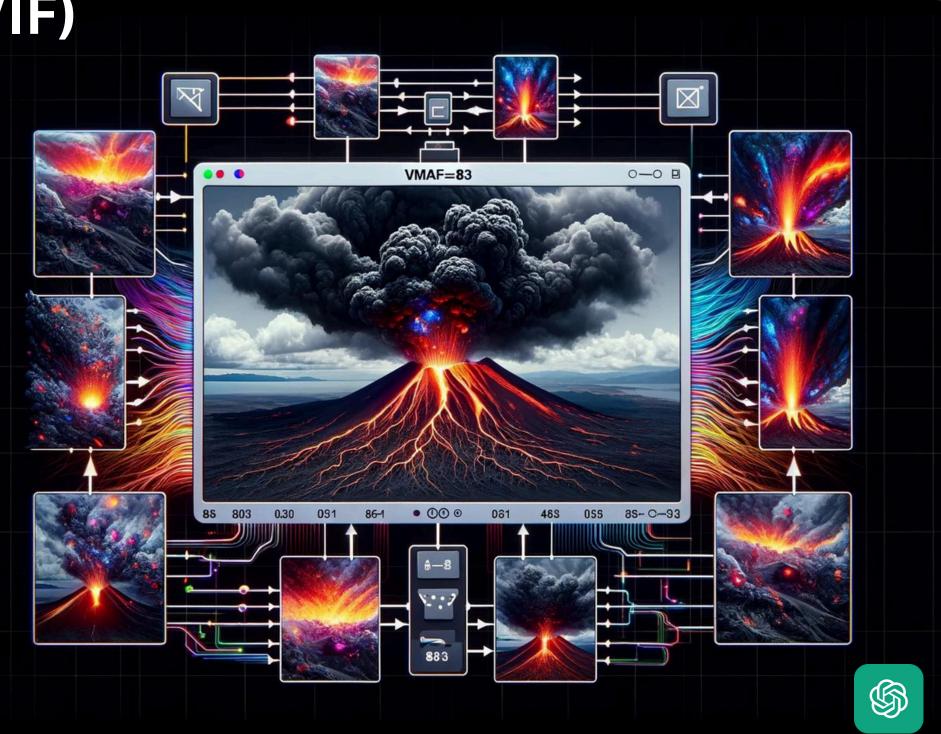




Video Multimethod Assessment Fusion (VMAF)

Fusion de plusieurs mesures de qualité

- Visual Information Fidelity (VIF)
- Detail Loss Metric (DLM)
- Mouvement (temporel)



Comment utiliser VMAF: Étape 0 préparer le contenu

```
>ffmpeg -i Sollevante_SDR_UHD_24fps.mov
-c:v libx264 -preset veryfast -b 1M
Sollevant_1mbps.mp4
```

Fichier source

Encodeur (x264)

Vitesse

Débit 1 Mbps

Ficher encodé





Comment utiliser VMAF: Étape 1 invoquer VMAF

```
>ffmpeg -i Sollevant_1mbps.mp4
-i Sollevante_SDR_UHD_24fps.mov
-lavfi libvmaf=log_path=Sollevant_1mbps.json:log_fmt='json'
-f null -
```

Ficher encodé

Fichier source

Fichier JSON

Spécifier le format JSON

Comment utiliser VMAF: Étape 2 extraire l'information

```
import json
with open('SolLevant_1mbps.json') as f:
    data = json.load(f)

vmaf = data["pooled_metrics"]["vmaf"]["harmonic_mean"]
```

Comment utiliser VMAF: Bonus obtenir plusieurs metriques

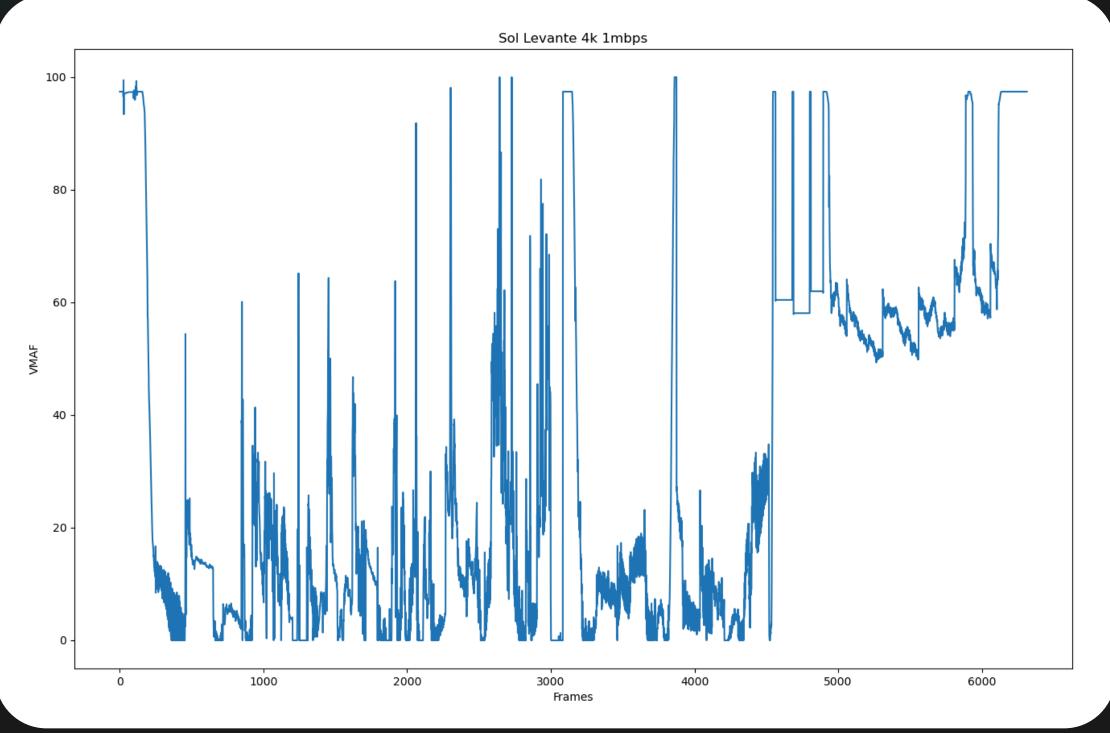
```
>ffmpeg -i SolLevant_1mbps.mp4
-i SolLevante_SDR_UHD_24fps.mov
-lavfi libvmaf=log_path=SolLevant_1mbps.json:log_fmt='json': \
    feature=name=psnrlname=float_ssim
-f null -
```

```
import json
with open('SolLevant_1mbps.json') as f:
    data = json.load(f)
print("vmaf=%0.2f psnr y=%0.2f cb=%0.2f cr=%0.2f ssim=%0.2f" %
        (data["pooled_metrics"]["vmaf"]["harmonic_mean"],
        data["pooled_metrics"]["psnr_y"]["mean"],
        data["pooled_metrics"]["psnr_cb"]["mean"],
        data["pooled_metrics"]["psnr_cr"]["mean"],
        data["pooled_metrics"]["float_ssim"]["mean"]))
```

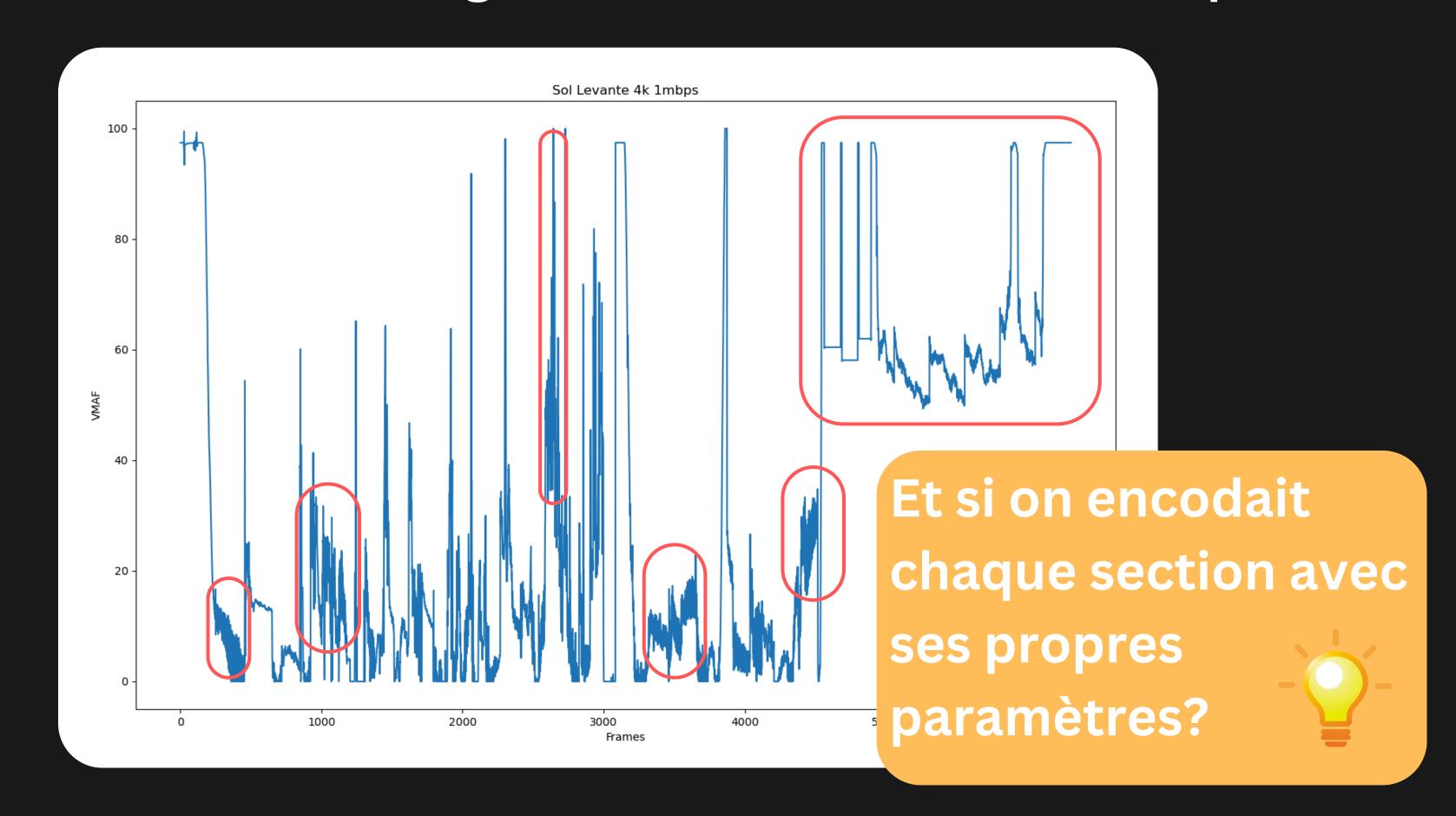
Comment utiliser VMAF: Bonus 2 faire un graphique

```
import matplotlib.pyplot as plt
import json
with open('SolLevant_lmbps.json') as f:
    data = json.load(f)

vmafs = []
for f in data["frames"]:
    vmafs.append(f["metrics"]["vmaf"])
plt.plot(vmafs)
plt.xlabel("Frames")
plt.ylabel("VMAF")
plt.show()
```



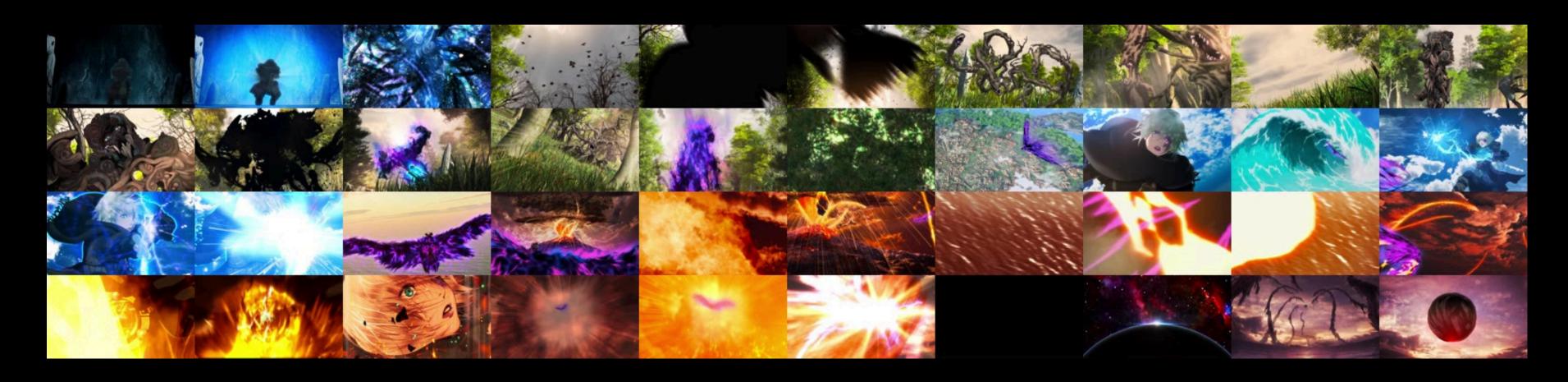
Les valeurs de VMAF changent selon les sections de la séquence



L'IA et l'encodage optimisé par le contenu

Encodage optimisé par le contenu

On sépare une séquence en plusieurs segments



Chaque segment est encodé avec des paramètres différents

Séparer une séquence en segments

Fichier source

Filtre de détection de changement de scène (différentiel > 40%)

Extraire les métadonnées dans un fichier



timestats.txt

```
frame:Ø pts:45400 pts_time:18.916667
lavfi.scene_score=0.759362
...
frame:10 pts:111400 pts_time:46.416667
lavfi.scene_score=0.533200
...
frame:40 pts:412300 pts_time:171.791667
lavfi.scene_score=0.498394
```

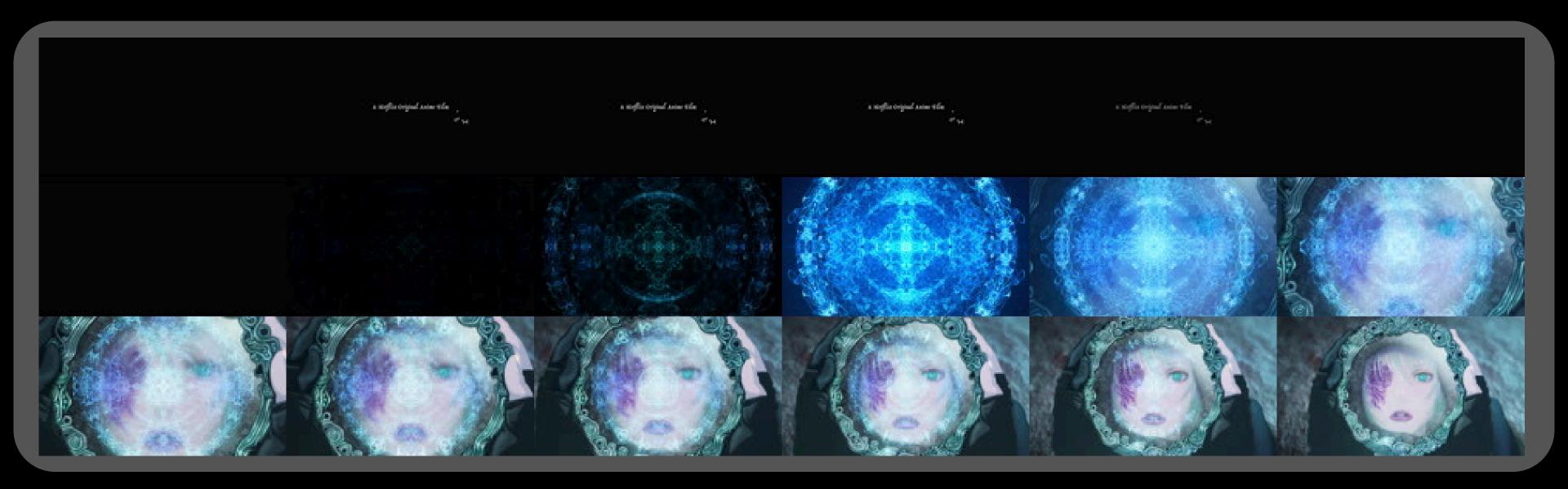
Séparer une séquence en segments

>ffmpeg -ss 0.0 -to 18.916667

-i SolLevante_SDR_UHD_24fps.mov

-vcodec ffv1 SolLevante_0.ffv1.mkv

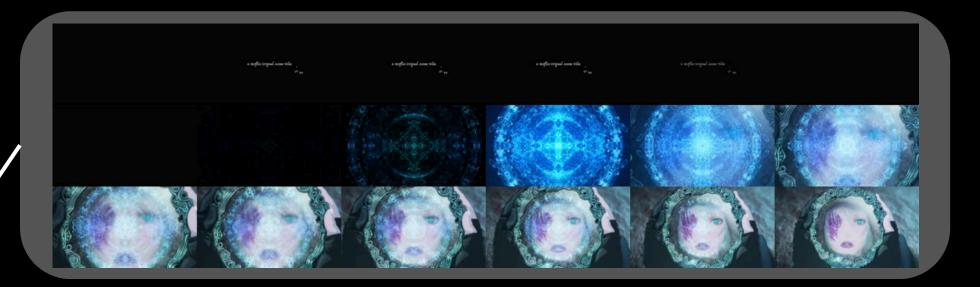
Temps de départ du segment
Temps de fin du segment
Fichier source
Encodage sans perte du segment
Fichier segment encodé



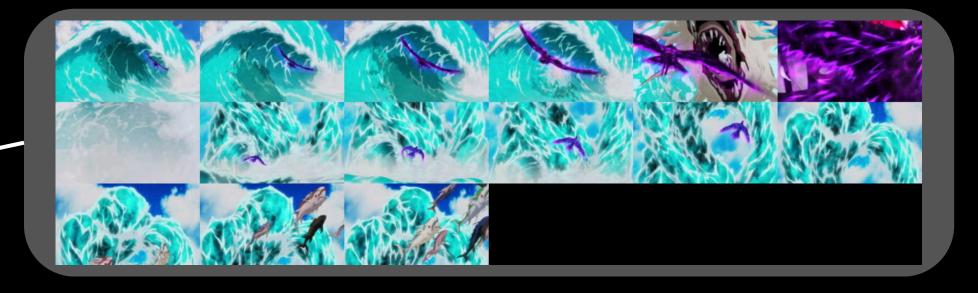
timestats.txt

lavfi.scene_score=0.422651

```
frame:Ø pts:45400 pts_time:18.916667
lavfi.scene_score=0.759362
...
frame:17 pts:179400 pts_time:74.75
lavfi.scene_score=0.413879
frame:18 pts:185200 pts_time:77.166667
lavfi.scene_score=0.473587
...
frame:25 pts:241300 pts_time:100.541667
```







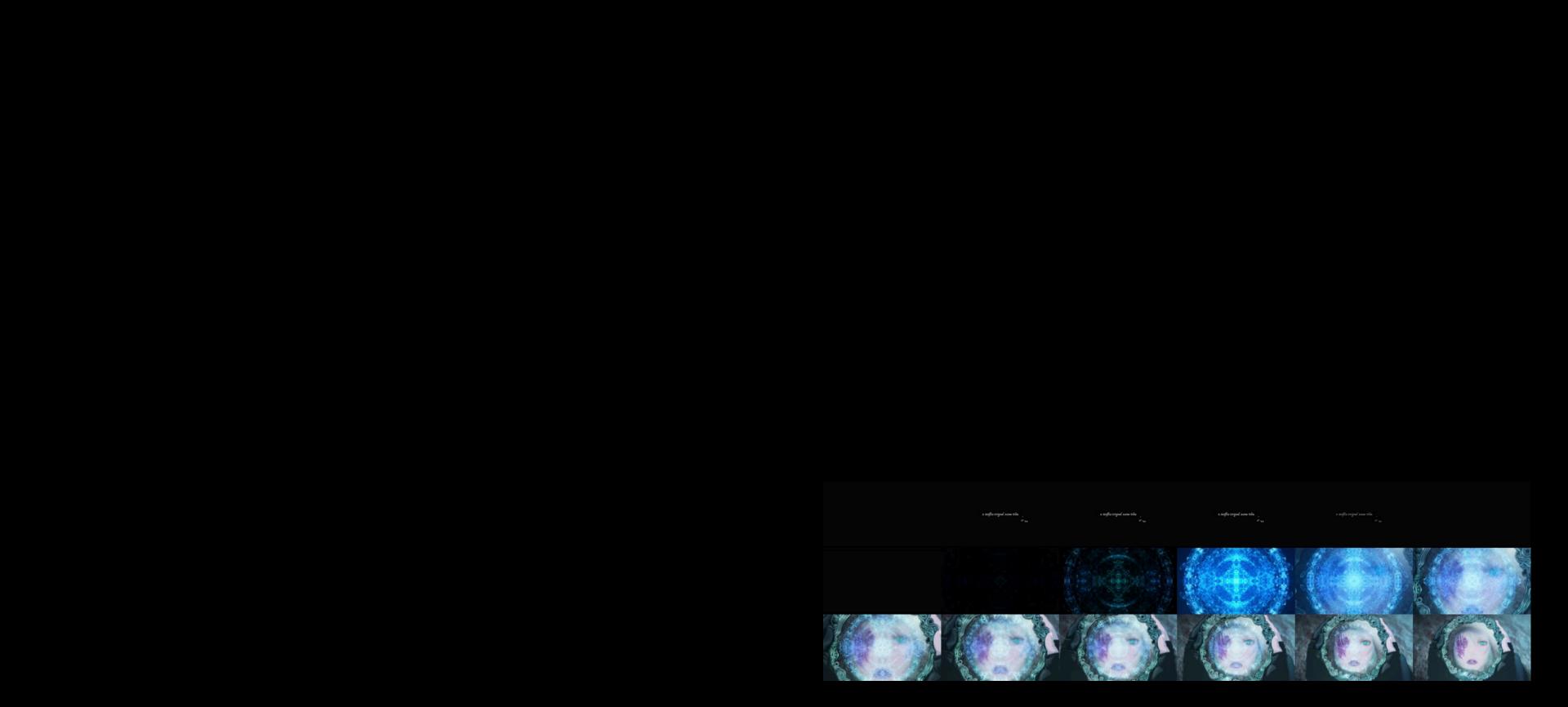


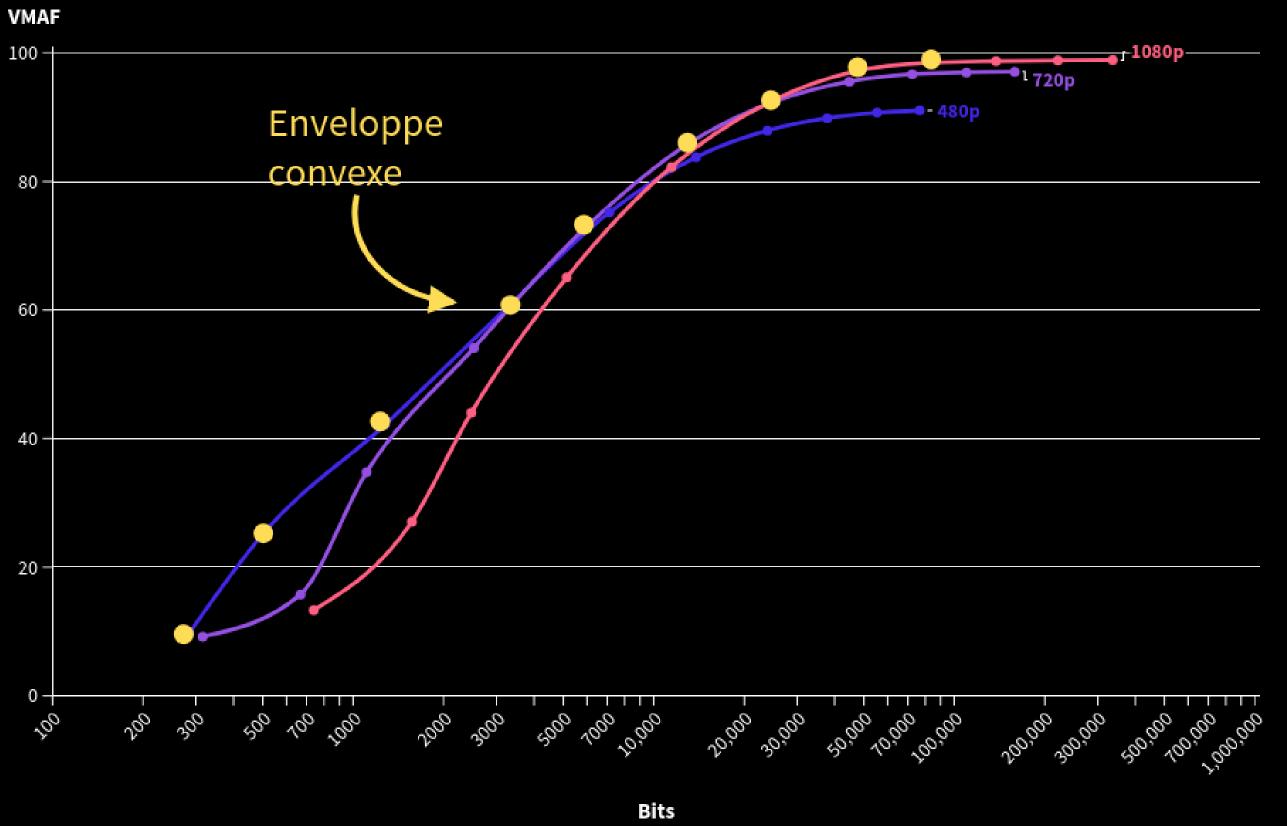
Encoder la séquence avec des paramètres différents

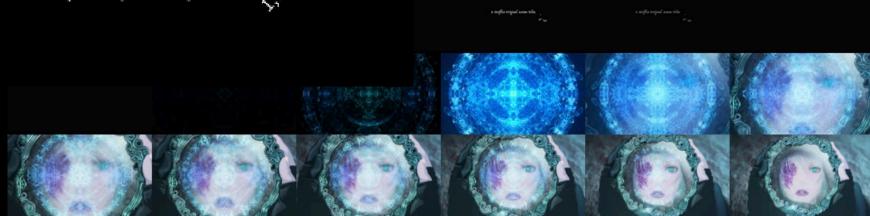
```
#!/bin/bash
mkdir encodes vmaf
for i in `seq 0 5 50`; do
   ffmpeg -i SolLevante 0.ffv1.mkv \
          -vf scale=854:480 \
          -c:v libx264 -preset ultrafast -crf $i encodes/SolLevante 0 $i.mkv
   ffmpeg -i encodes/SolLevante_0_$i.mkv -i SolLevante_0.ffv1.mkv \
          -filter complex "[0:v]scale=1920:1080[dist];" \
                          "[1:v]scale=1920:1080[ref];" \
                          "[dist][ref]libvmaf=log path=vmaf/SolLevant 0 $i.xml" \
          -f null -
   VMAF=$(grep -Eo 'vmaf.+harmonic_mean="(\d+\.\d+)"' vmaf/SolLevant_0 $i.xml | cut -d "=" -f5)
   SIZE=$(du -k "encodes/SolLevante_0_$i.mkv" | cut -f1)
   echo $SIZE,$VMAF >> SolLevante 0.csv
done
```

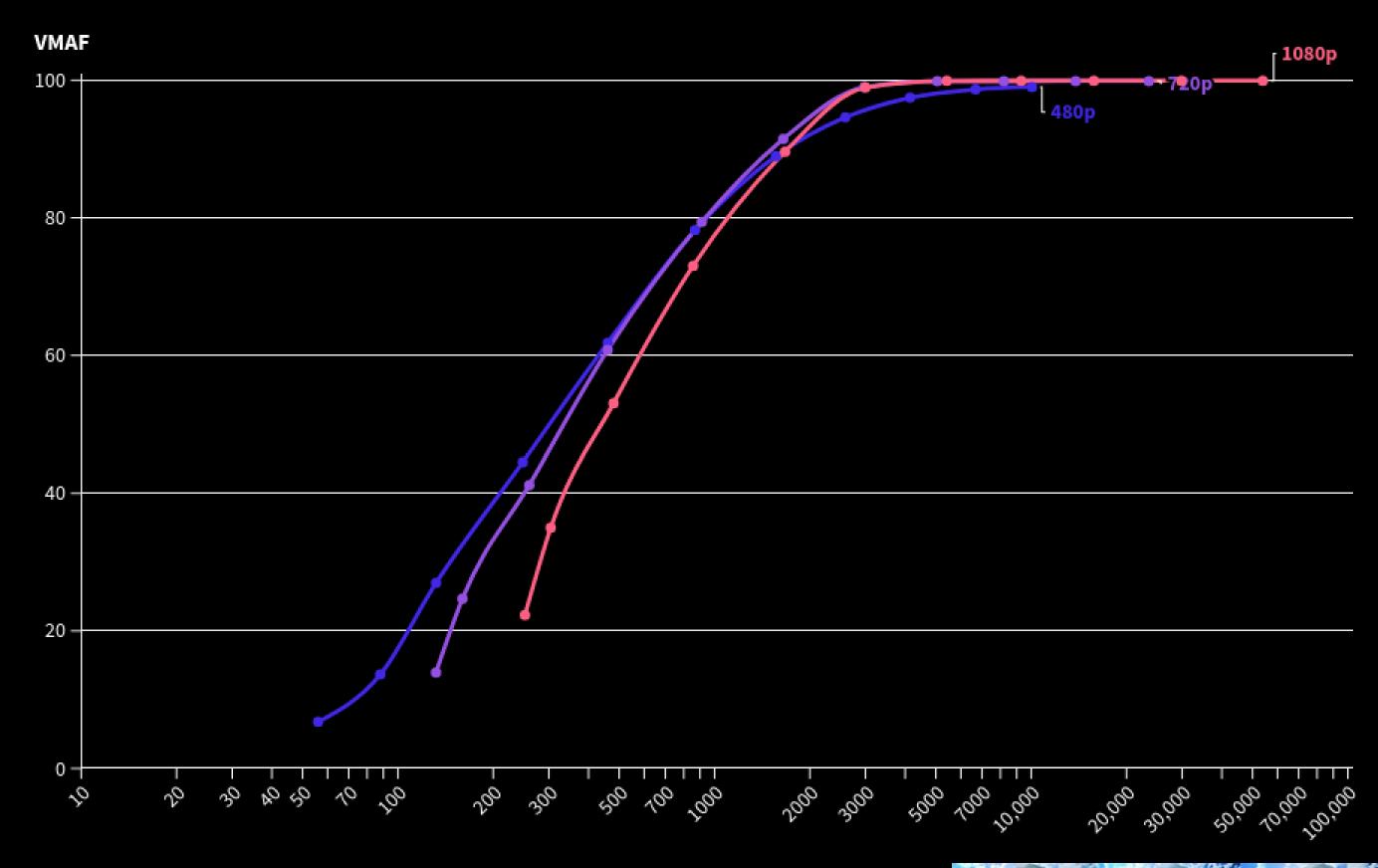
SolLevante_0.csv

```
76664, "91.Ø45787"
76664, "91.Ø45787"
553ØØ, "9Ø.7Ø295Ø"
37736, "89.847182"
23912, "87.9235Ø7"
13812, "83.775198"
712Ø, "75.2Ø4187"
3232, "6Ø.2755Ø6"
```

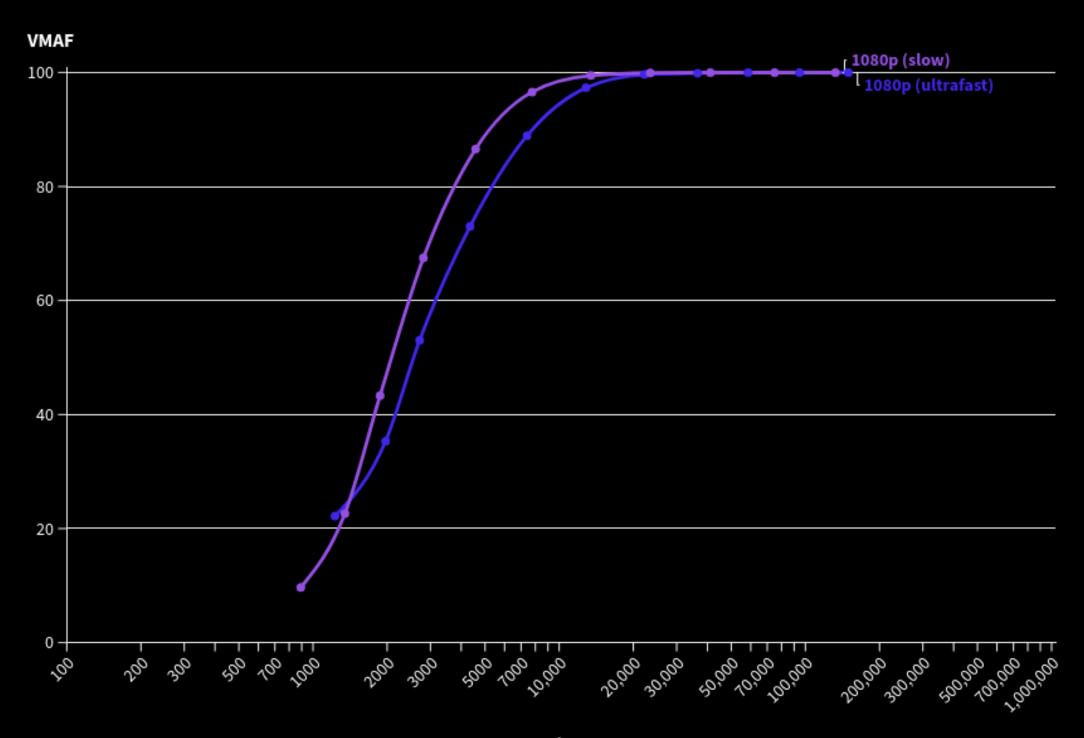








Mais où est l'IA?

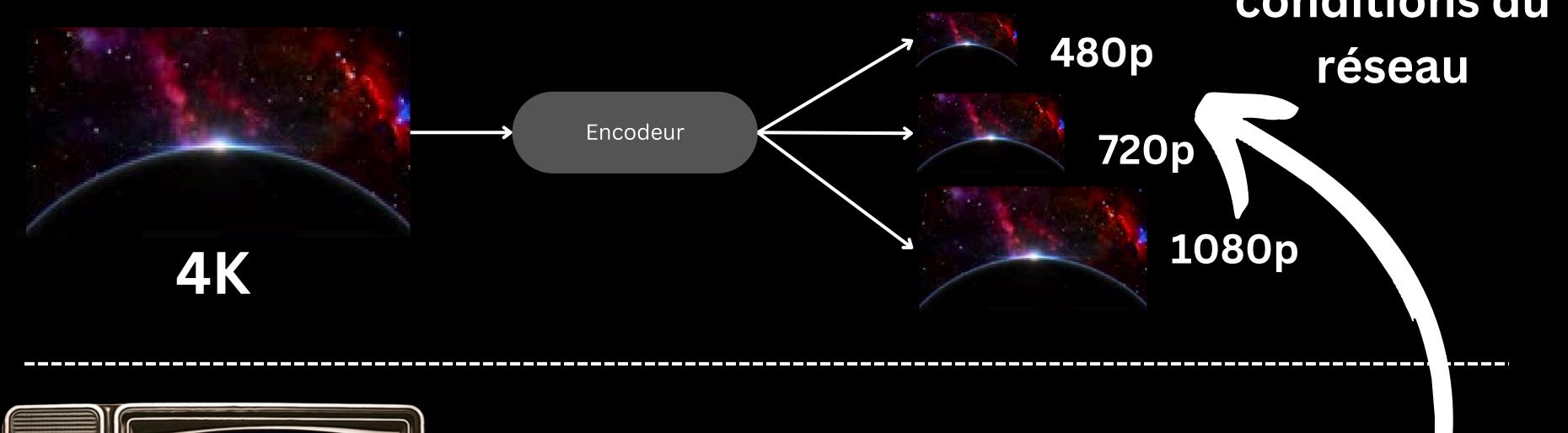


Utiliser un encodeur plus rapide donne une très bonne prédiction

L'IA et la super resolution

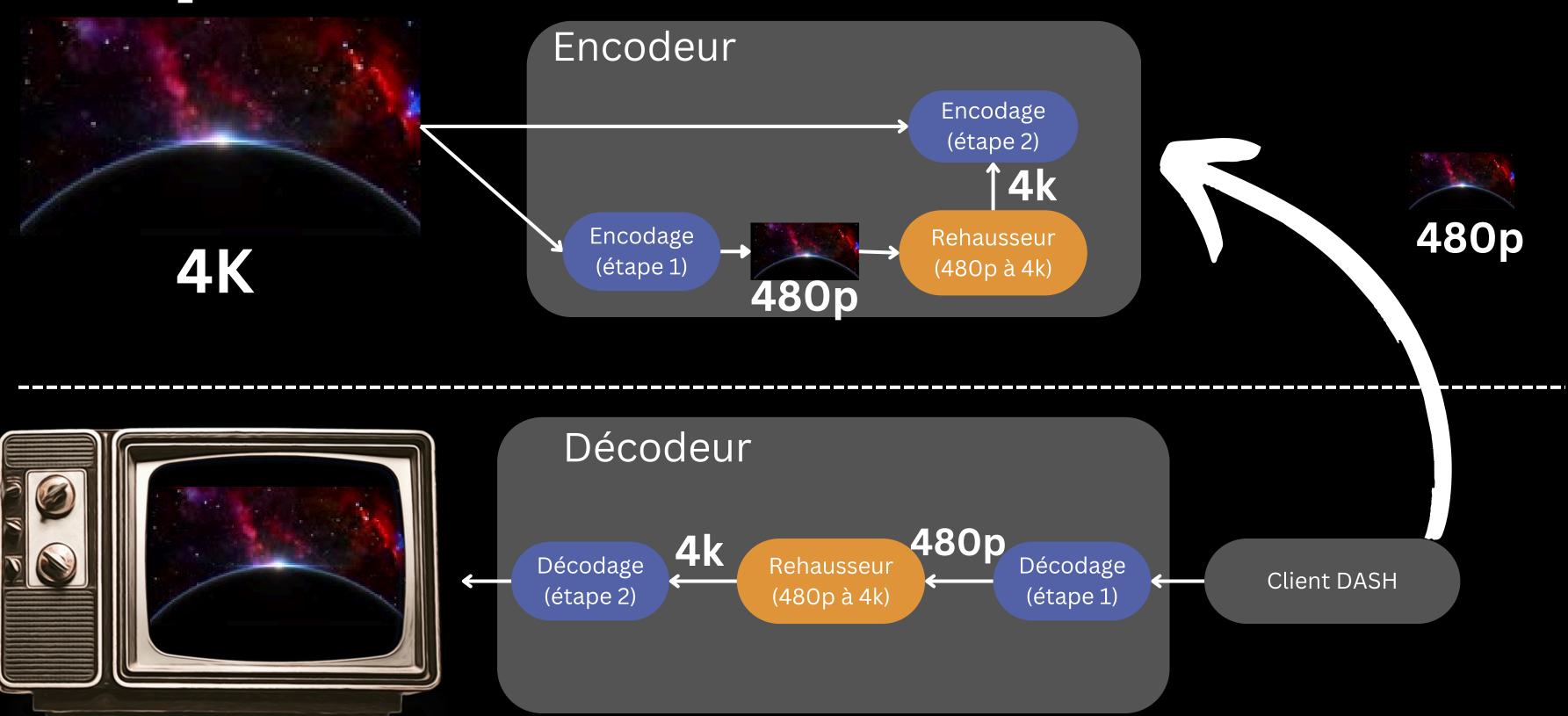
Situation actuelle

Selon les conditions du réseau





Super Resolution (AV1)



Mot de la fin



En conclusion, ne choisissez pas une technologie simplement parce qu'elle utilise l'I.A. Comme le dit le dicton, "l'habit ne fait pas le moine". Analysez chaque produit pour s'assurer qu'il répond vraiment à vos besoins. La vraie valeur se trouve dans sa capacité à résoudre vos problèmes et à apporter des améliorations concrètes.