

# Projet GPGPU

Fall 2024 - GISTRE&SCIA S9

# Objectif

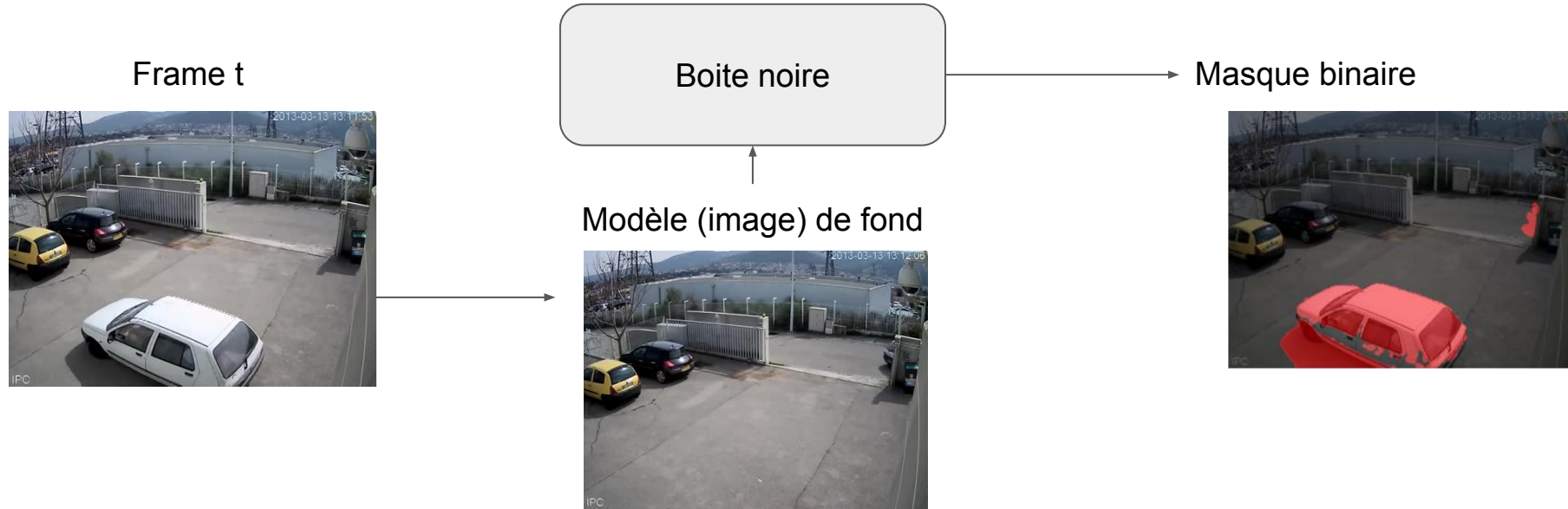
Plugin GStreamer de séparation fond / objets mobiles dans des vidéos

Étape préliminaire dans de nombreuses chaînes de traitement

Beaucoup d'opération locales  $\Rightarrow$  Bon candidat pour une optimisation GPU



# Calcul du masque de changement

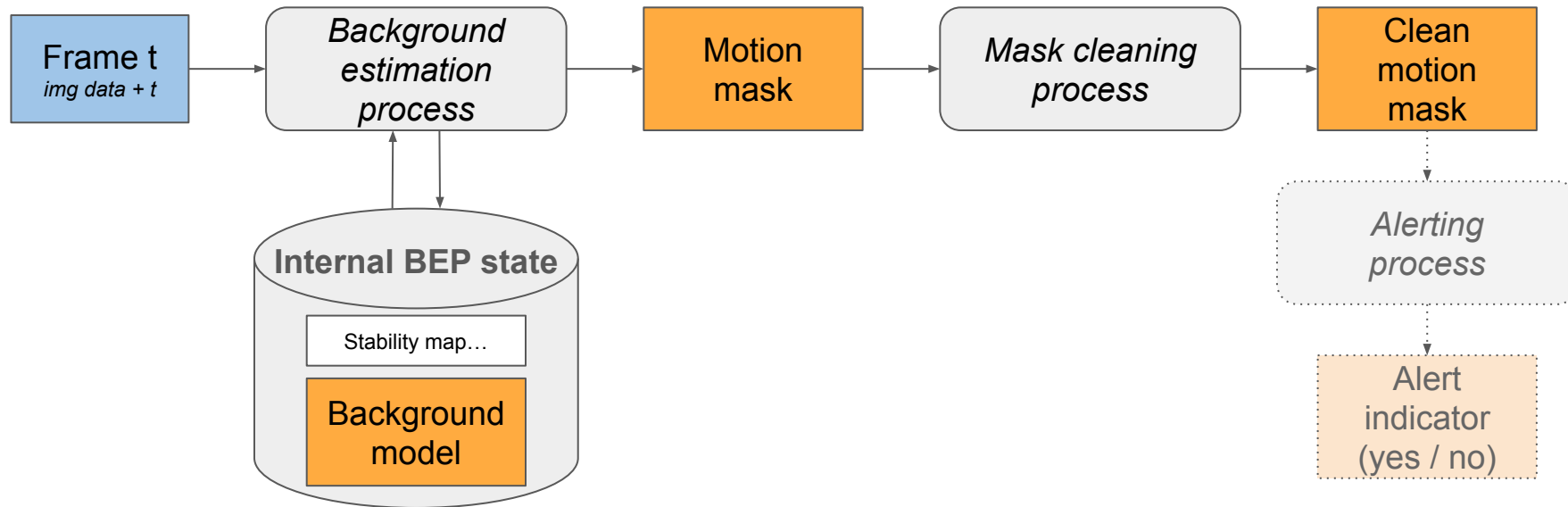


Input  
data

Output  
data

Process

# Approche générale

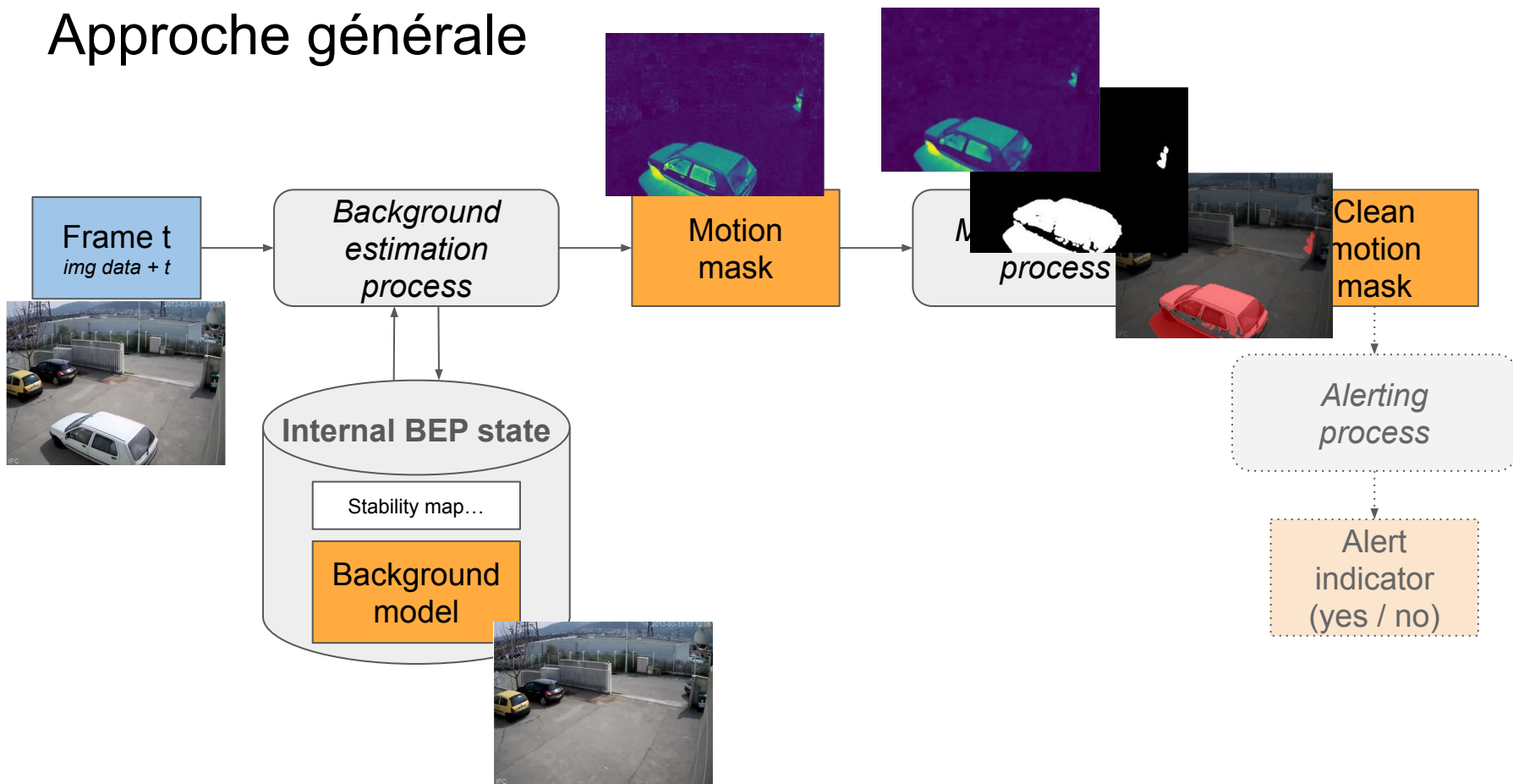


Input  
data

Output  
data

Process

# Approche générale



# Détail des étapes

# 1. Background estimation process

## *Pixel-wise process, for each frame*

**States:** background value, candidate value, time since last match (t)

**Init:** bg value  $\leftarrow$  current pixel value

match = distance(bg, frame) < 25 (using Lab color space)

If we do not have a match:

if time == 0

candidate value = current value; t++

else if time < 100

average candidate value and current value; t++

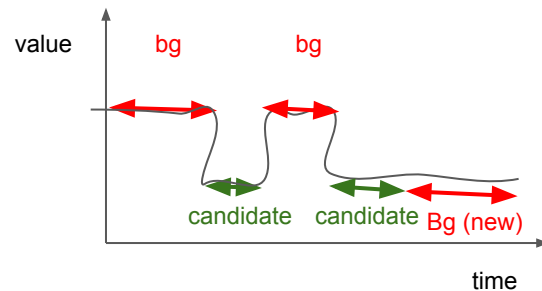
else

switch candidate and background; t=0

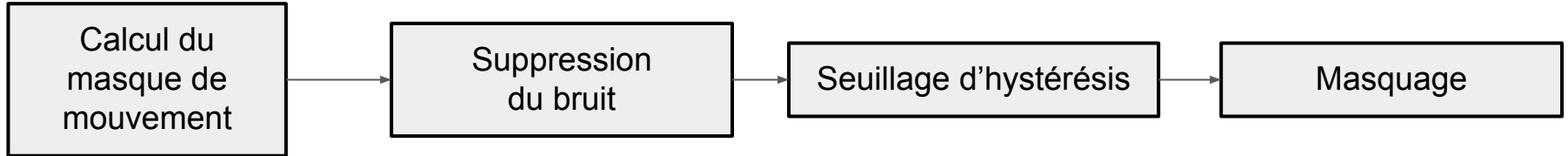
else: (we have a match)

average background value and current value; t=0

return match distance



## 2. Mask cleaning process overview



*Résultat du processus précédent.*

*C'est une carte qui indique pour chaque pixel un score de mouvement  $\geq 0$ .*



*Ouverture morphologique par un disque de rayon 3.*

*La taille du rayon doit être ajusté à la taille de l'image.*



*Seuillage d'hystérésis.  
Seuil bas: 4  
Seuil haut: 30*

*La valeur de seuil dépend de l'espace colorimétrique.*

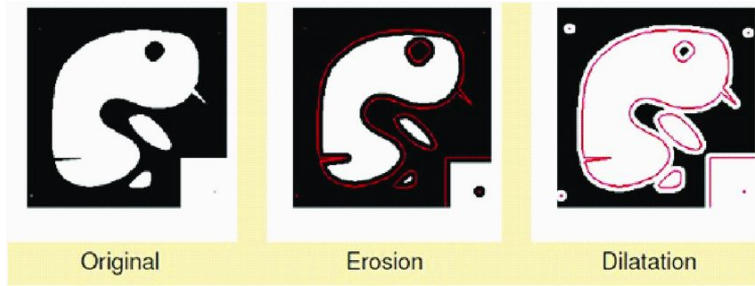


*$\text{input} + 0.5 * \text{red} * \text{masque}$*



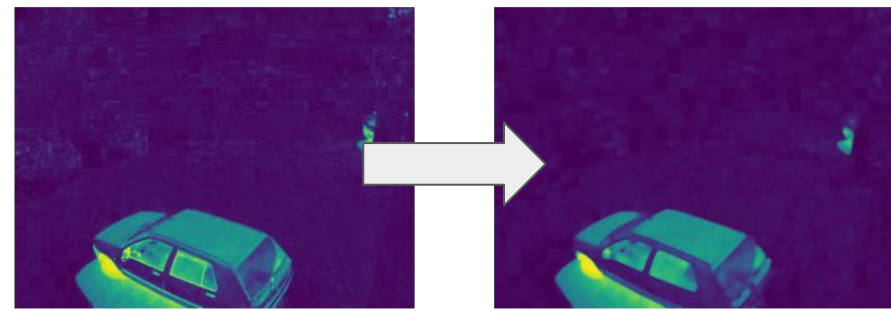
## 2.1 Suppression du bruit

Ouverture morphologique (érosion suivi d'une dilatation)



[https://en.wikipedia.org/wiki/Mathematical\\_morphology](https://en.wikipedia.org/wiki/Mathematical_morphology)

[https://en.wikipedia.org/wiki/Opening\\_\(morphology\)](https://en.wikipedia.org/wiki/Opening_(morphology))



Erosion:

new value for  $p(x,y)$  = **min** value in neighborhood of  $p(x,y)$

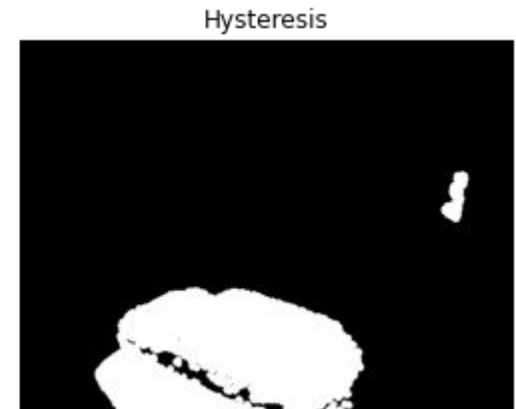
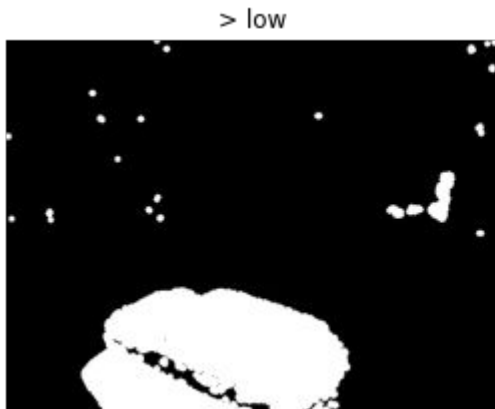
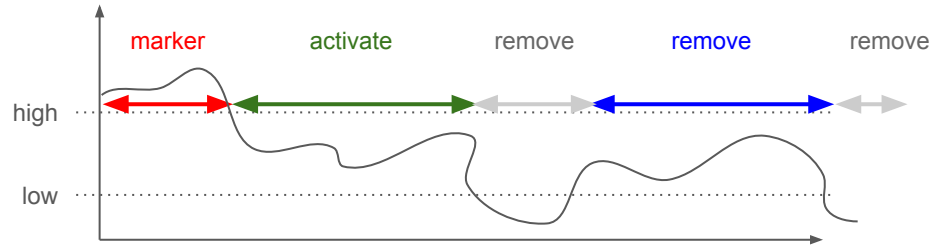
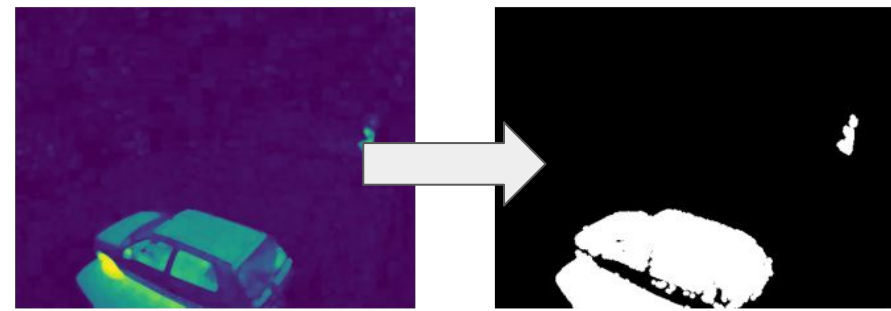
Dilation:

new value for  $p(x,y)$  = **min** value in neighborhood of  $p(x,y)$

## 2.2. Seuillage d'hystérésis

Principe :

- Supprimer les signaux faibles
- Propager les signaux forts vers les signaux moyens



## 2.2. Implémentation de la reconstruction d'hystérésis

Idée:

On propage les pixels des marqueurs dans le masque jusqu'à stabilité.

Les marqueurs sont initialisés avec les éléments supérieurs au seuil haut.

L'entrée contient tous les éléments supérieurs au seuil bas.

```
__device__ bool has_changed;

__global__ void reconstruction(input, marker, out) {
    int p = .... + threadIdx.x;
    if (out[p] || !input[p]) // already processed or too low
        return;
    if (marker[p]) { // initialize output with markers
        out[p] = true;
        has_changed = true;
        return;
    }
    for (int q : neighbors(p)) // Is any of my neighbors active?
        if (out[q]) { // propagate neighbor activation to me
            out[p] = true;
            has_changed = true;
        }
    }
}

int main() {
    out = black_image;
    has_changed = false;
    while (has_changed)
        reconstruction<<<>>>(input, marker, out);
}
```

Matériel fourni

# Code gstreamer

- Code de base fourni sur Moodle
- Vous devez implémenter un filtre GStreamer CUDA et CPP

Idéalement, en intégrant les paramètres suivants :

```
bg=uri           : uri vers une image de fond (default="" => estimé)
opening_size=(int) : taille de l'ouverture
th_low=(int)      : valeur basse du filtre (default=3)
th_high=(int)     : valeur haute du filtre (default=30)
```

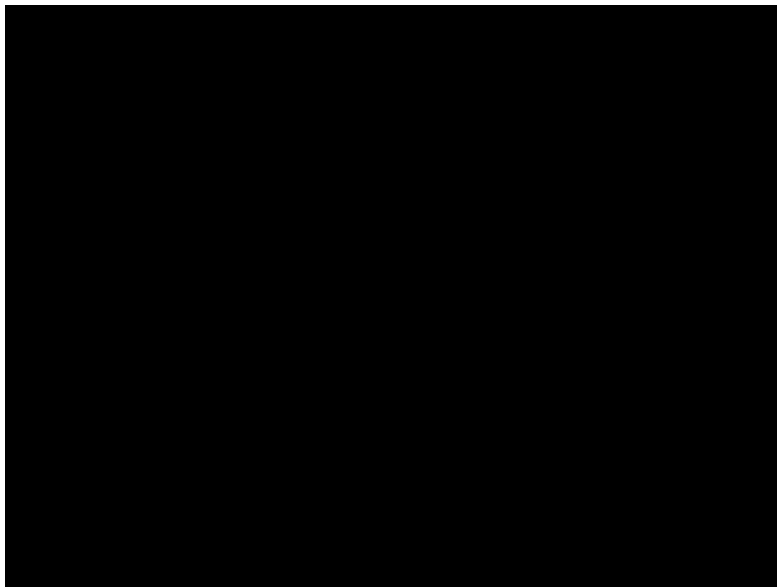
```
bg_sampling_rate=(int) : intervalle d'échantillonnage des frames pour l'estimation de fond
                        (default=500ms)
bg_number_frame=(int)  : nombre de frames utilisées pour l'estimation de fond (default=10)
```

# Code gstreamer

- Code de base fourni sur Moodle
- Vous devez implémenter un filtre GStreamer CUDA et CPP ayant pour paramètres:

```
bg=uri           : uri vers une image de fond (default="" => estimé)
opening_size=(int) : taille de l'ouverture
th_low=(int)      : valeur basse du filtre (default=3)
th_high=(int)     : valeur haute du filtre (default=30)
```

```
bg_sampling_rate=(int) : intervalle d'échantillonnage des frames pour l'estimation de fond
                        (default=500ms)
bg_number_frame=(int)  : nombre de frames utilisées pour l'estimation de fond (default=10)
```



```
gst-launch-1.0 uridecodebin uri=file:///tmp/sintel_trailer-480p.webm ! videoconvert ! "video/x-raw,  
format=(string)RGB" ! cudafilter ! videoconvert ! video/x-raw, format=I420 ! x264enc ! mp4mux ! filesink  
location=video.mp4
```

Demo

Attendus



# Critères d'évaluation

1. **Code correct**  $\Rightarrow$  résultats ACCEPTABLES au niveau qualitatif  
*Avec cette méthode, les résultats ne seront pas optimaux*
2. **Vitesse**  $\Rightarrow$  plus le framerate est rapide, mieux c'est.

# Conseils

1. **Avoir une version C++ fonctionnelle**  $\Rightarrow$  baseline
2. **Git tag des versions du programme**  $\Rightarrow$  permet de mesurer plusieurs versions/optimisations du programme (+ conserver variantes pour comparaison)
3. **Faire les optimisations une par une**  $\Rightarrow$  permet d'évaluer les optimisations réellement significatives de façon individuelle

# Livrables

## 1. Implémentation

- Source code for C++ CPU reference
- Source code for CUDA implementation(s)
- Source code for benchmark tools
- Build scripts (GNU Make, CMake...)

Nous devons être capables de reproduire vos résultats.

## 2. Rapport succinct

- Description du sujet
- Répartition des tâches par membre du groupe
- Benchmarks et graphiques des performances des versions (CPU + GPU + GPU Optimisé ##)
- Analyse des performances et des bottlenecks (graphiques nsight / nvprof)

## 3. Slides de la soutenance

## 4. Répartition des groupes (sur Moodle) => 30 oct. (pour les affectations de soutenance)

# Soutenances

19-20 et 28  
nov

- 15' présentation
- 5' démo
  - *Données:* <https://cloud.lrde.epita.fr/s/xZXGCa8dMX3GsTW>
- 5' discussion

Soutenances sur Teams.

Projet par groupe de 4.

Tous les membres du groupe doivent être présents à la soutenance.

Vous devez rendre tous les fichiers le 18 nov. au soir et être inscrit·e à votre groupe.