

# TP4

## Objectifs :

- Traitement d'image (partie 1)

## Ressources

- <http://docs.nvidia.com/cuda/index.html>
- [https://docs.nvidia.com/cuda/pdf/CUDA\\_C\\_Programming\\_Guide.pdf](https://docs.nvidia.com/cuda/pdf/CUDA_C_Programming_Guide.pdf)

## Setup

- haswell-cuda ou 192.168.80.64
- Login : nom.prenom (minuscule)
- Passwd : toto (à changer)
- Vérification bashrc, PATH et LD\_LIBRARY\_PATH

**Git :** <https://github.com/hpc-apps/Cuda.git>

## Exercice 1

- Récupérer le programme grayscale\_init.cpp
- Remplir tous les pixels à la valeur zero
  - `cv::Mat m_out( m_in.rows, m_in.cols, CV_8UC1, g.data() )`
- Compiler cet exemple  
`g++ -o exe grayscale_init.cpp $(pkg-config --libs --cflags opencv)`

## Exercice 2

- La conversion en niveau de gris suit la formule ci-dessous. Compléter le code fourni afin de convertir une image en niveau de gris (pour chaque pixel)

$$\text{Grey} = (307 * \text{Red} + 604 * \text{Green} + 113 * \text{Blue}) / 1024$$

## Exercice 3

- Créer le code CUDA équivalent permettant de réaliser cette conversion
- Utiliser d'abord une grille 1D puis une grille 2D de threads

## Exercice 4

- Instrumenter les codes CPU et GPU afin de comparer les temps d'exécution (évaluer l'impact de différentes tailles de blocs pour la version CUDA)