

L'obiettivo di questa esercitazione è di confrontarsi con la programmazione di un'applicazione che sfrutti la GPU usando il toolkit C for CUDA per GPU NVIDIA.

L'algoritmo da tradurre è un filtro per immagini che esegue, per ogni pixel dell'immagine, la media dei pixel circostanti nel raggio di 8 pixel.

## Prerequisiti

1. Scheda grafica NVIDIA
2. CUDA Toolkit 10.0 installato – <https://developer.nvidia.com/cuda-downloads>
3. Codice sorgente di partenza ed immagine di test (*average\_filter.zip*)

## NOTA

Chi non ha a disposizione un PC con scheda grafica NVIDIA non avrà la possibilità di eseguire e verificare il codice programmato.

## Esercizio – Primi passi

Il primo programma che dovrete sviluppare si occuperà di trasferire l'immagine verso la GPU e dalla GPU nuovamente verso l'host.

Modificate a tale scopo la funzione *average\_gpu(...)* contenuta nel file sorgente *average\_gpu.cu* in modo da eseguire la copia come richiesto.

## Esercizio – Primi kernel

Seguendo l'implementazione dell'algoritmo di *average\_cpu(...)* contenuto del file *average\_cpu.c* introducete i seguenti 3 kernel:

- *read\_channel* legge dall'immagine di input un determinato canale
- *compute\_average* esegue la copia pixel per pixel di un canale in input ad uno in output
- *write\_channel* esegue la scrittura di un determinato canale nell'immagine di output

## Esercizio – Versione GPU dell'algoritmo

Completate il kernel *compute\_average* creato al punto precedente introducendo la logica necessaria ad eseguire l'algoritmo di *average\_cpu(..)* contenuto del file *average\_cpu.c*.