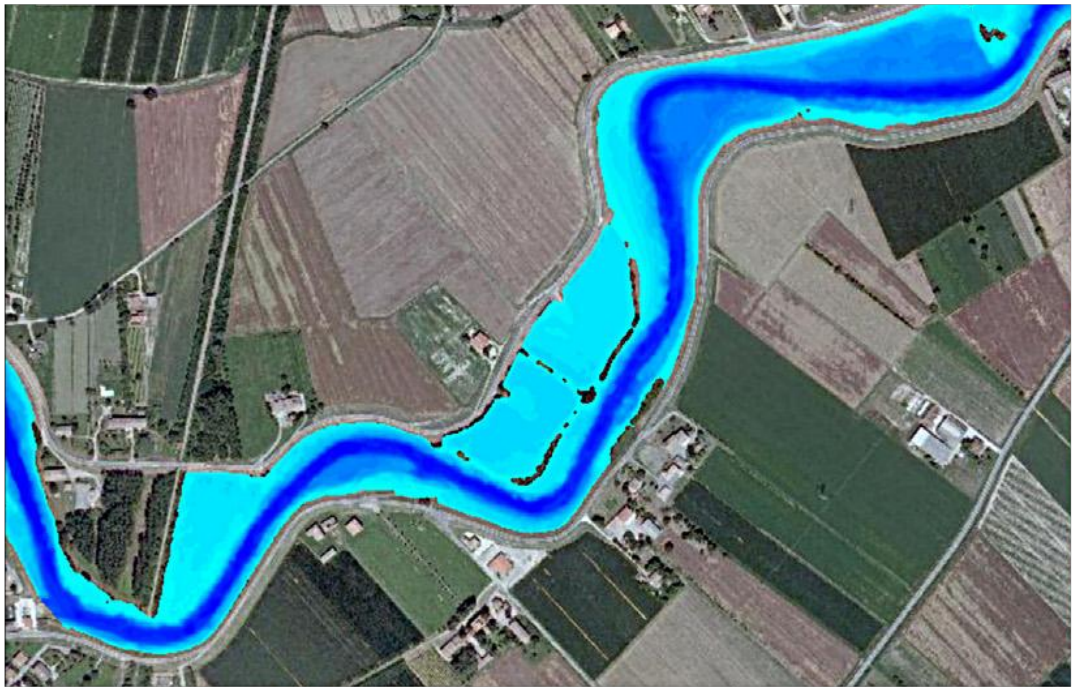
Studio di algoritmi per la rappresentazione di dati geografici con multi risoluzione

Pietralberto Mazza

27 Ottobre 2016

Indice

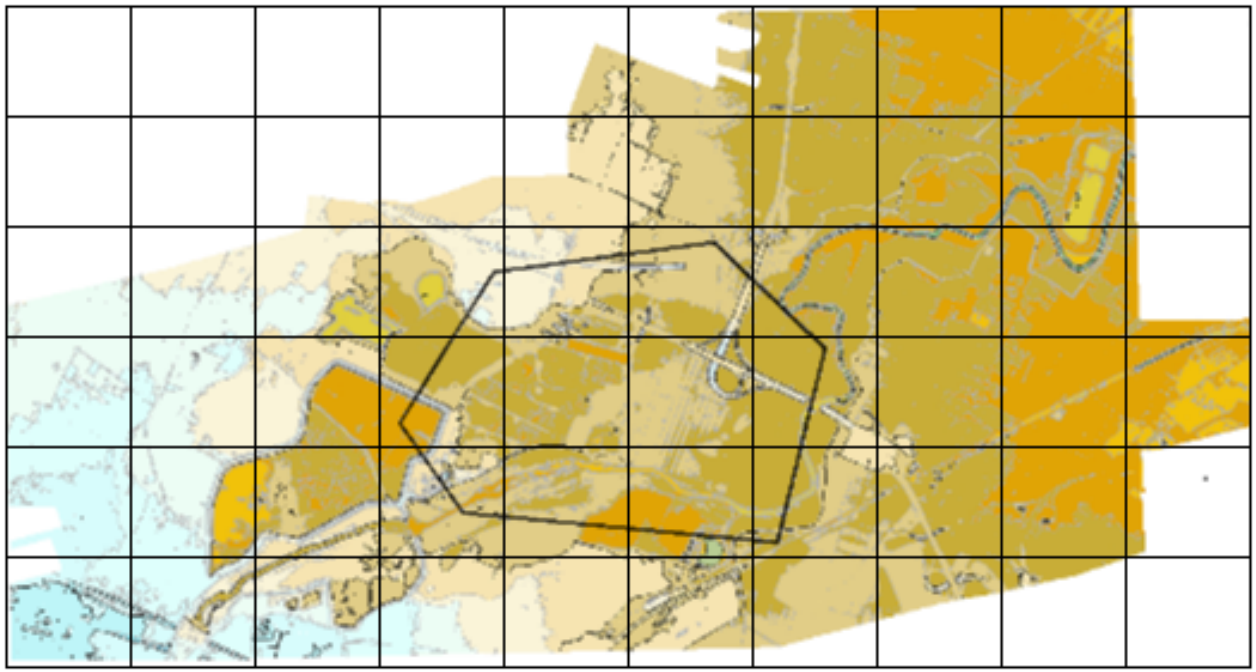
Introduzione



- Vengono compiute simulazioni d'acqua su mappe geografiche (batimetrie)
- Ogni punto della mappa esprime l'altezza del terreno

Introduzione

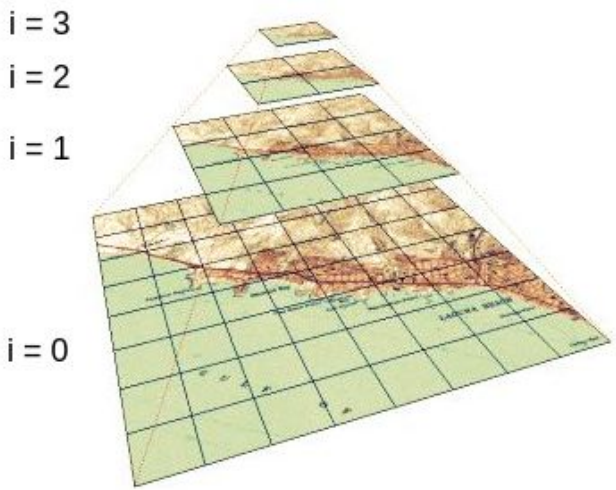
Mappa e Poligono

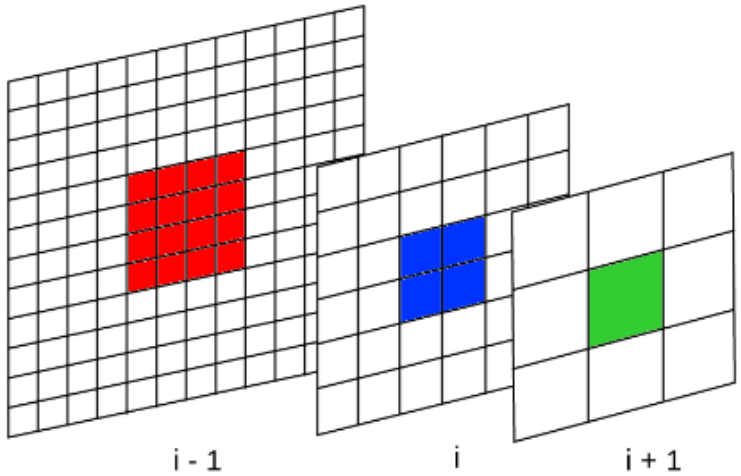


- Una mappa è composta da più tavolette
- Sulla mappa viene definita un'area su cui compiere la simulazione

Introduzione

Rappresentazione dei livelli di risoluzione

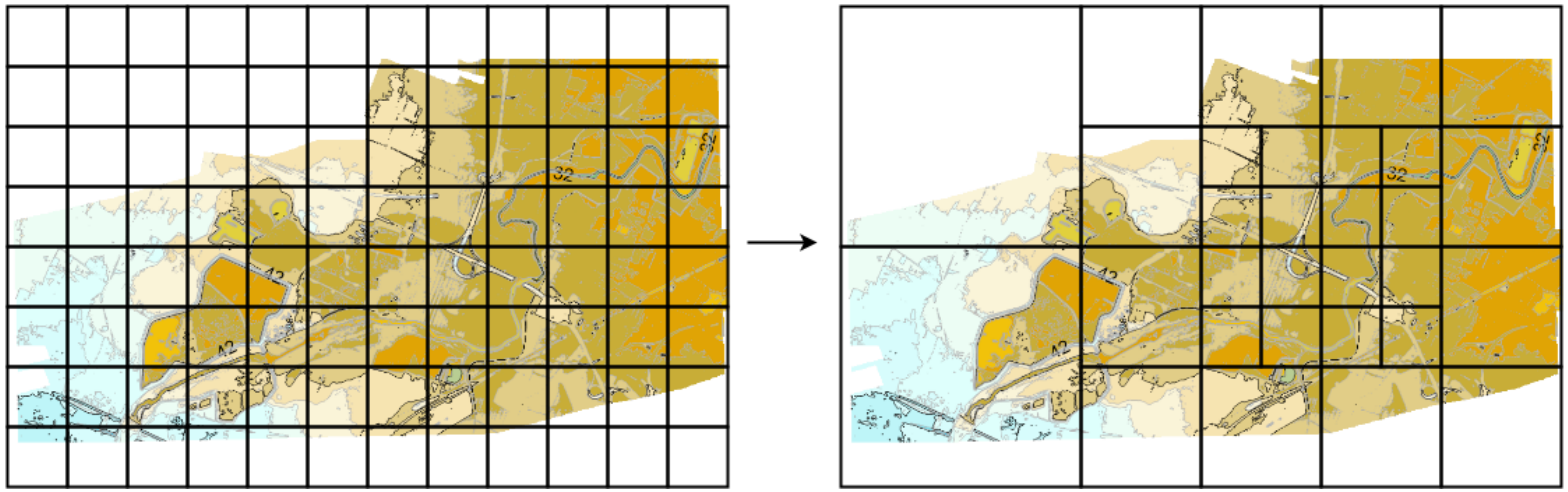




Per ogni livello di risoluzione viene utilizzata una bitmask

Introduzione

Rappresentazione in multi risoluzione

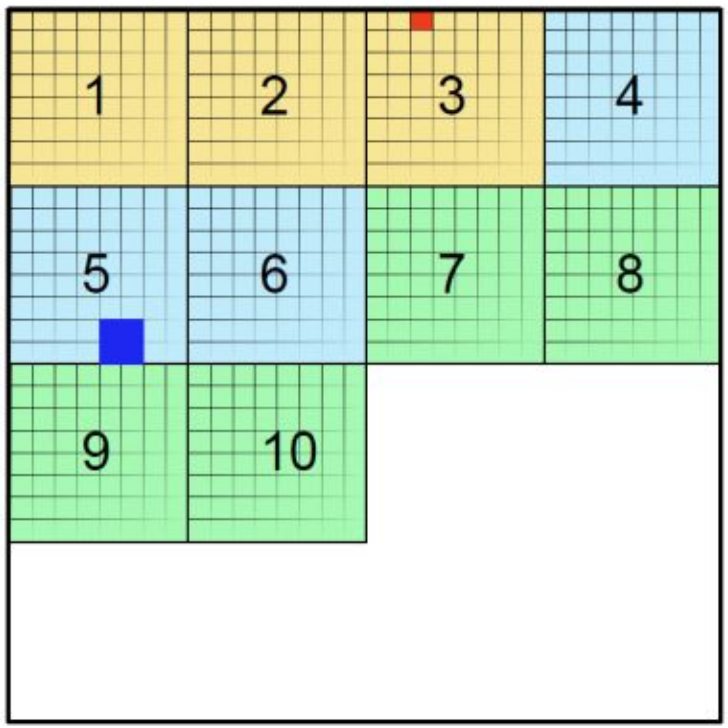


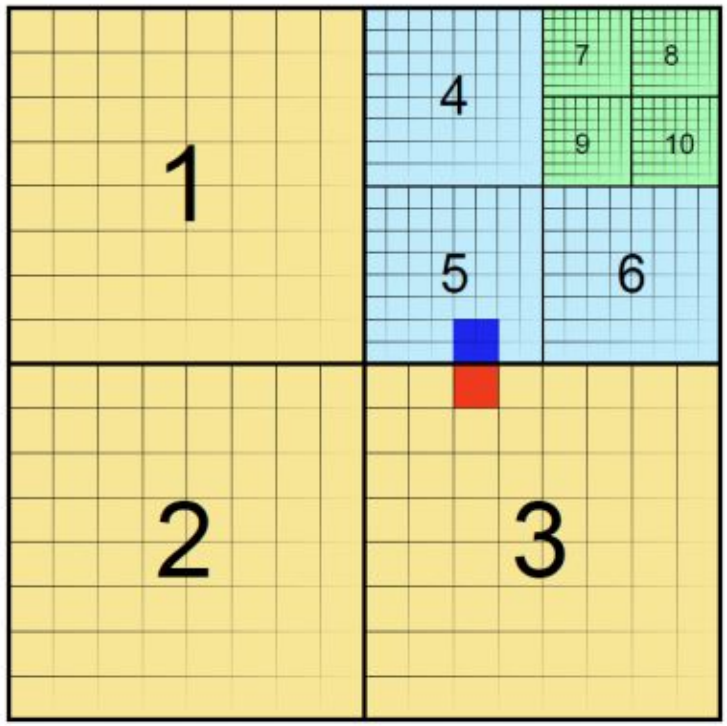
Le mappe vengono rappresentate in multi risoluzione per ridurre l'impatto in memoria

Il programma viene eseguito su GPU (Graphics Processing Unit)

Introduzione

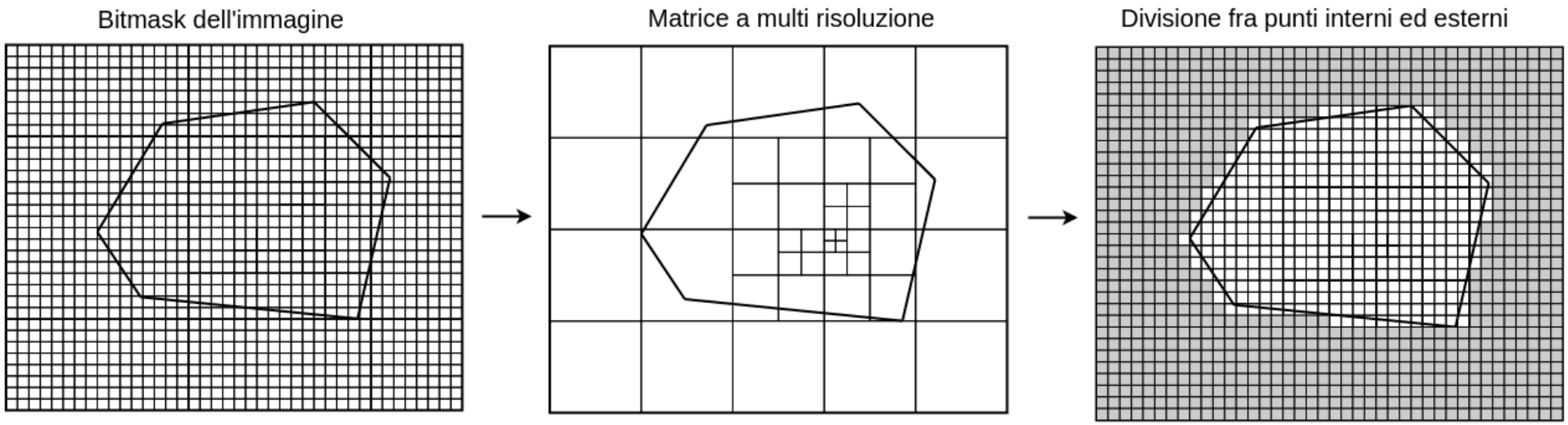
Matrice a multi risoluzione





Una matrice a multi risoluzione è una matrice di blocchi codificati che vengono ordinati in base al numero

Stato dell'arte



Viene allocata troppa memoria

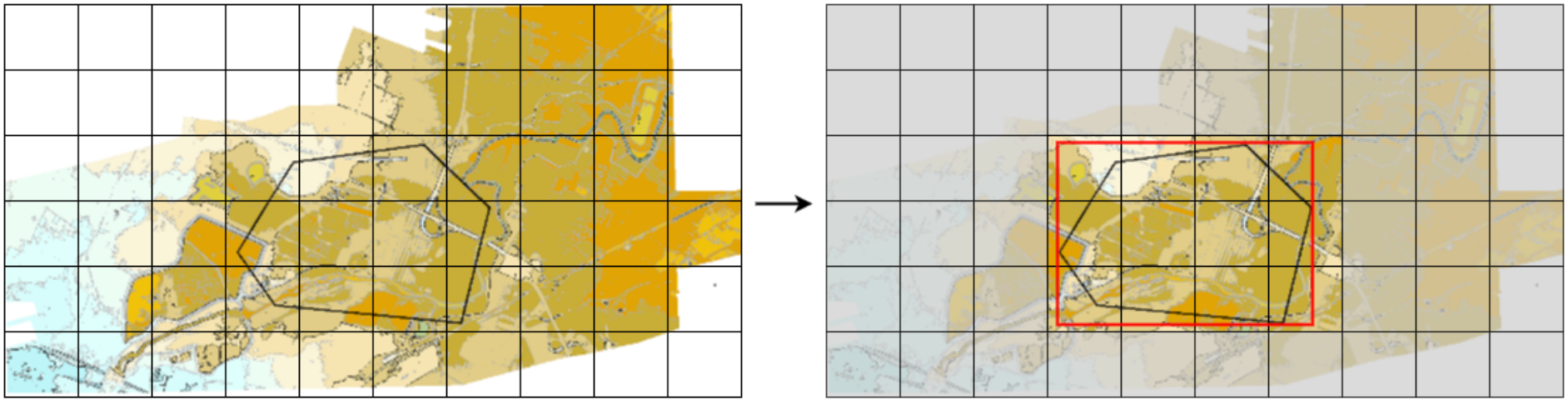
Obiettivo della tesi

Ridurre l'impatto in memoria

- Bitmask dell'immagine
- Matrice a multi risoluzione
- Caricare i dati delle altezze allocando memoria solo per una tavoletta

Realizzazione

Bounding box



Viene individuata la sottomatrice di tavolette in cui cade il poligono

Realizzazione

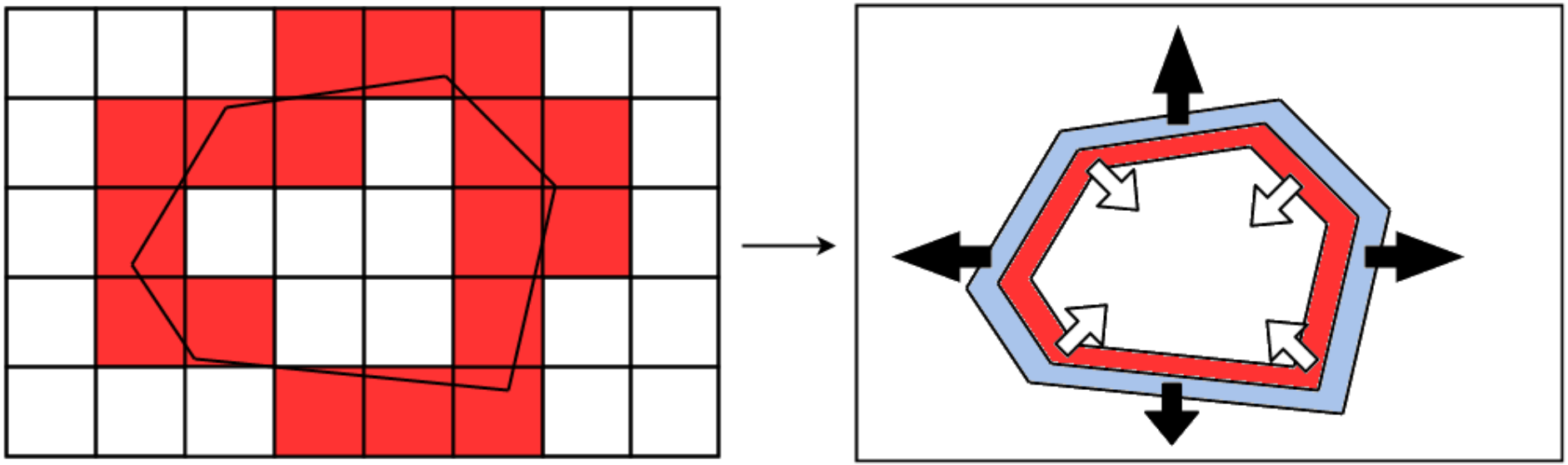
Generazione multi risoluzione

immagine

La multi risoluzione viene generata partendo dai punti che interpolano i segmenti del poligono

Realizzazione

Divisione fra punti interni ed esterni



- Vengono identificati i blocchi in cui sono contenuti i seed points
- La divisione viene eseguita utilizzando due code ed esplorando i vicini

Realizzazione

Caricamento delle altezze

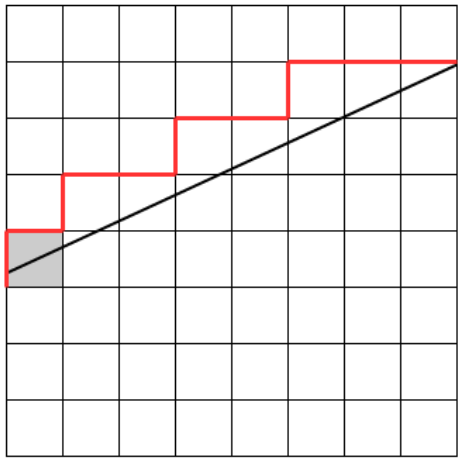
load_data()

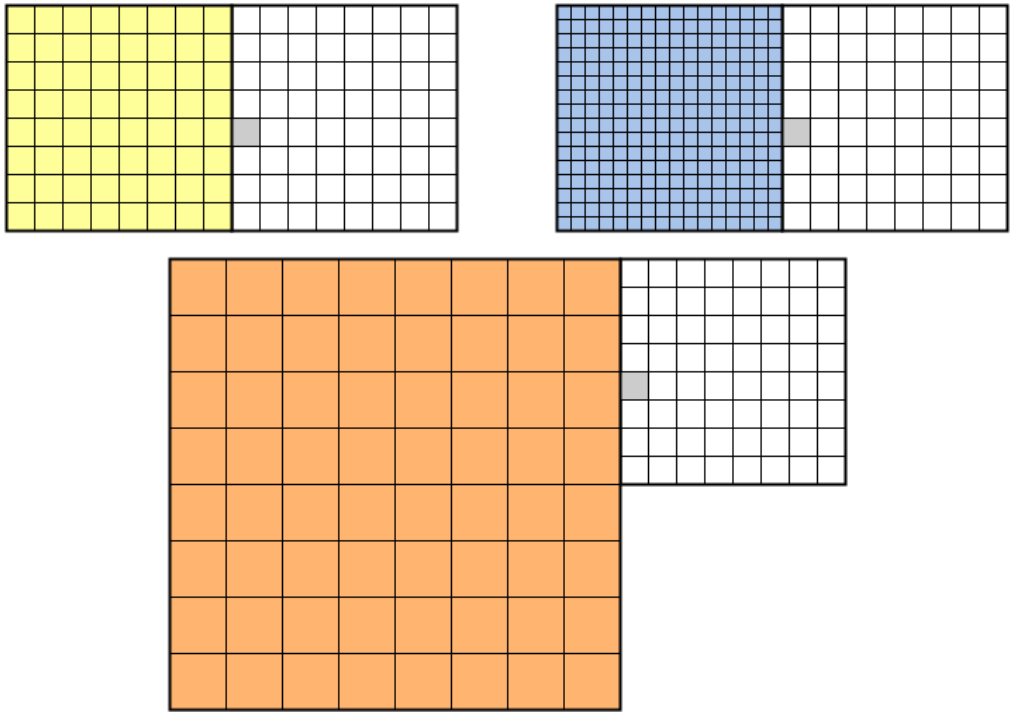
```
for each slab in bounding_box
    for each height in slab
        p = real_point(height)
        b = find_block_in_grid_multi(p)
        b.value += height
        ++c[b]
    end
end

for each b in grid_multi
    if c[b] != 0
        b.value /= c[b]
```

Realizzazione

Differenze fra le due versioni

- Le dimensioni della matrice a multi risoluzione dipendono dal poligono
- Bitmask dell'immagine in multi risoluzione
- Per ogni cella sul bordo di un blocco il vicino può essere a risoluzione diversa





Risultati

Passaggi della suddivisione fra punti interni ed esterni di un bounding block

Conclusioni

Livello 3

Bounding box	Celle originali	Celle multires	Memoria
pol1.bln	9072000	212992	31MB
pol2.bln	5292000	131072	24MB
pol3.bln	12096000	278528	35MB
pol4.bln	567000	32768	17MB
bln_secchia.bln	18900000	409600	46MB
bln_stretto.bln	2646000	73728	20MB

Livello 2

Bounding box	Celle originali	Celle multires	Memoria
pol1.bln	9072000	409600	46MB
pol2.bln	5292000	253952	34MB
pol3.bln	12096000	475984	51MB
pol4.bln	567000	90112	21MB
bln_secchia.bln	18900000	638976	63MB
bln_stretto.bln	2646000	163840	27MB

Idealmente la matrice a multi risoluzione può pesare 12GB (memoria della GPU), ovvero $7,5 * 10^9$ celle rappresentanti un'area di 86*86 km^2

Sviluppi futuri

- Integrazione del progetto nel sistema già esistente
- Caricare i dati solo per i punti interni al poligono
- Aggiungere e adattare nella nuova versione l'algoritmo per l'eliminazione dei punti con 3 vicini esterni
- Aggiungere le condizioni di muro

Grazie per l'attenzione