Laboratorul 5

# Analiza cerințelor

**Proiect CUDA: *Convoluția pe imagini (matrice bidimensionale de pixeli).*** Convoluția se referă la procesul de adăugare a vecinilor la fiecare pixel al unei imagini, cu ponderile date de un kernel (altfel spus o mască, adică o matrice de dimensiuni n×n, unde n este impar și de obicei n ≥ 3).

# Proiectare

Funcția KERNEL CUDA, și anume cea care se execută pe device, pe fiecare thread:

* folosește un array bidimensional de blocuri, cu dimensiunea:
* folosește un array bidimensional de threaduri, cu dimensiunea:

,

* folosește memorie shared alocată dinamic pentru threadurile din același bloc, având dimensiunea de mai sus
* algoritmul de convoluție folosit:

**set** acumulator **to** zero

**for each** linie de elemente **in** kernel:

**for each** element **in** linie de elemente:

**set** rezulat **to multiply** element **to** valoarea pixelului inițial corespunzător

**add** rezultat **to** acumulator

**set** pixel din rezultat **to** acumulator

Pixelul inițial folosit în calcul, cât și fiecare pixel corespunzător pozițiilor din kernel sunt calculate local, folosind indicii de poziție ale threadurilor și blocurilor.

Convoluția se realizează doar în cazul în care threadul curent se referă la o poziție care nu este pe bordură.

# Detalii de implementare

Deoarece suprapunerea kernelului centrat peste pixelii de pe marginea pozei ar însemna necesitatea unor valori inexistente din afara limitelor imaginii, au fost propuse diferite metode pentru rezolvarea acestei probleme. Rezolvarea aleasă de noi în implementare este de a considera valorile inexistente ca având valoarea 0.

# Cazuri de testare

Testarea a fost realizată folosind blocuri de dimensiuni 4×4, respectiv 16×16, folosind următoarele date de input și obținând rezultatul:

* Kernel, dimensiune 3×3:

|  |  |  |
| --- | --- | --- |
| 1 | 1 | 1 |
| 1 | 2 | 1 |
| 1 | 1 | 1 |

* Matrice inițială, dimensiune 16×16:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 7 | 7 | 2 | 6 | 0 | 6 | 4 | 1 | 3 | 0 | 2 | 8 |
| 4 | 5 | 6 | 7 | 3 | 4 | 0 | 0 | 3 | 1 | 4 | 7 | 7 | 7 | 8 | 6 |
| 8 | 9 | 10 | 11 | 8 | 5 | 0 | 3 | 3 | 4 | 4 | 0 | 8 | 2 | 2 | 1 |
| 5 | 1 | 5 | 8 | 2 | 2 | 1 | 8 | 4 | 2 | 3 | 5 | 7 | 6 | 5 | 2 |
| 2 | 2 | 3 | 8 | 0 | 7 | 6 | 3 | 8 | 2 | 1 | 0 | 7 | 4 | 8 | 4 |
| 3 | 2 | 0 | 5 | 4 | 5 | 8 | 1 | 6 | 5 | 5 | 6 | 3 | 8 | 5 | 2 |
| 5 | 6 | 3 | 7 | 5 | 3 | 1 | 3 | 3 | 1 | 3 | 1 | 3 | 4 | 8 | 5 |
| 6 | 8 | 0 | 2 | 6 | 8 | 8 | 3 | 4 | 4 | 2 | 3 | 5 | 4 | 4 | 0 |
| 6 | 5 | 8 | 5 | 3 | 3 | 1 | 4 | 3 | 7 | 5 | 0 | 3 | 1 | 2 | 1 |
| 4 | 4 | 5 | 7 | 2 | 7 | 8 | 3 | 2 | 5 | 0 | 6 | 6 | 4 | 0 | 2 |
| 2 | 0 | 6 | 8 | 3 | 3 | 4 | 3 | 0 | 0 | 2 | 0 | 6 | 0 | 7 | 6 |
| 3 | 5 | 4 | 1 | 3 | 1 | 2 | 4 | 0 | 0 | 4 | 3 | 5 | 4 | 7 | 2 |
| 4 | 0 | 0 | 3 | 8 | 0 | 4 | 2 | 1 | 6 | 1 | 0 | 7 | 4 | 1 | 7 |
| 4 | 8 | 4 | 5 | 4 | 6 | 3 | 3 | 7 | 2 | 0 | 4 | 4 | 6 | 8 | 7 |
| 7 | 1 | 5 | 7 | 3 | 1 | 2 | 5 | 7 | 7 | 6 | 4 | 2 | 1 | 4 | 5 |
| 6 | 3 | 8 | 8 | 8 | 1 | 7 | 5 | 8 | 6 | 4 | 6 | 6 | 3 | 0 | 5 |

* Matrice rezultat, dimensiune 16×16:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 10 | 19 | 26 | 31 | 38 | 30 | 21 | 17 | 16 | 24 | 27 | 27 | 28 | 27 | 33 | 32 |
| 31 | 50 | 60 | 64 | 58 | 40 | 27 | 17 | 29 | 30 | 35 | 45 | 42 | 46 | 44 | 33 |
| 40 | 62 | 72 | 71 | 58 | 30 | 23 | 25 | 31 | 32 | 34 | 45 | 57 | 54 | 41 | 25 |
| 32 | 46 | 62 | 63 | 53 | 33 | 36 | 44 | 41 | 33 | 24 | 40 | 46 | 55 | 39 | 24 |
| 17 | 25 | 37 | 43 | 41 | 42 | 47 | 48 | 47 | 38 | 30 | 37 | 53 | 57 | 52 | 30 |
| 23 | 28 | 36 | 40 | 48 | 44 | 45 | 40 | 38 | 39 | 29 | 35 | 39 | 58 | 53 | 34 |
| 35 | 39 | 36 | 39 | 50 | 51 | 41 | 40 | 33 | 34 | 33 | 32 | 40 | 48 | 48 | 29 |
| 42 | 55 | 44 | 41 | 48 | 46 | 42 | 33 | 36 | 36 | 28 | 28 | 29 | 38 | 33 | 20 |
| 39 | 51 | 52 | 43 | 46 | 49 | 46 | 40 | 38 | 39 | 37 | 30 | 35 | 30 | 20 | 10 |
| 25 | 44 | 53 | 54 | 43 | 41 | 44 | 31 | 29 | 29 | 25 | 34 | 32 | 33 | 23 | 20 |
| 20 | 33 | 46 | 47 | 38 | 36 | 39 | 29 | 17 | 13 | 22 | 32 | 40 | 39 | 39 | 30 |
| 17 | 29 | 31 | 37 | 33 | 29 | 25 | 24 | 16 | 14 | 20 | 31 | 34 | 45 | 45 | 32 |
| 28 | 32 | 30 | 35 | 39 | 31 | 29 | 28 | 26 | 27 | 21 | 28 | 44 | 50 | 47 | 39 |
| 28 | 41 | 37 | 44 | 41 | 37 | 29 | 37 | 47 | 39 | 30 | 32 | 36 | 43 | 51 | 39 |
| 36 | 47 | 54 | 59 | 46 | 36 | 35 | 52 | 57 | 54 | 45 | 40 | 38 | 35 | 43 | 34 |
| 23 | 33 | 40 | 47 | 36 | 23 | 28 | 39 | 46 | 44 | 37 | 34 | 28 | 19 | 18 | 19 |