

Progettazione di filtri di convoluzione separabili su piattaforma embedded

Relatore:
Prof. Stefano Mattoccia

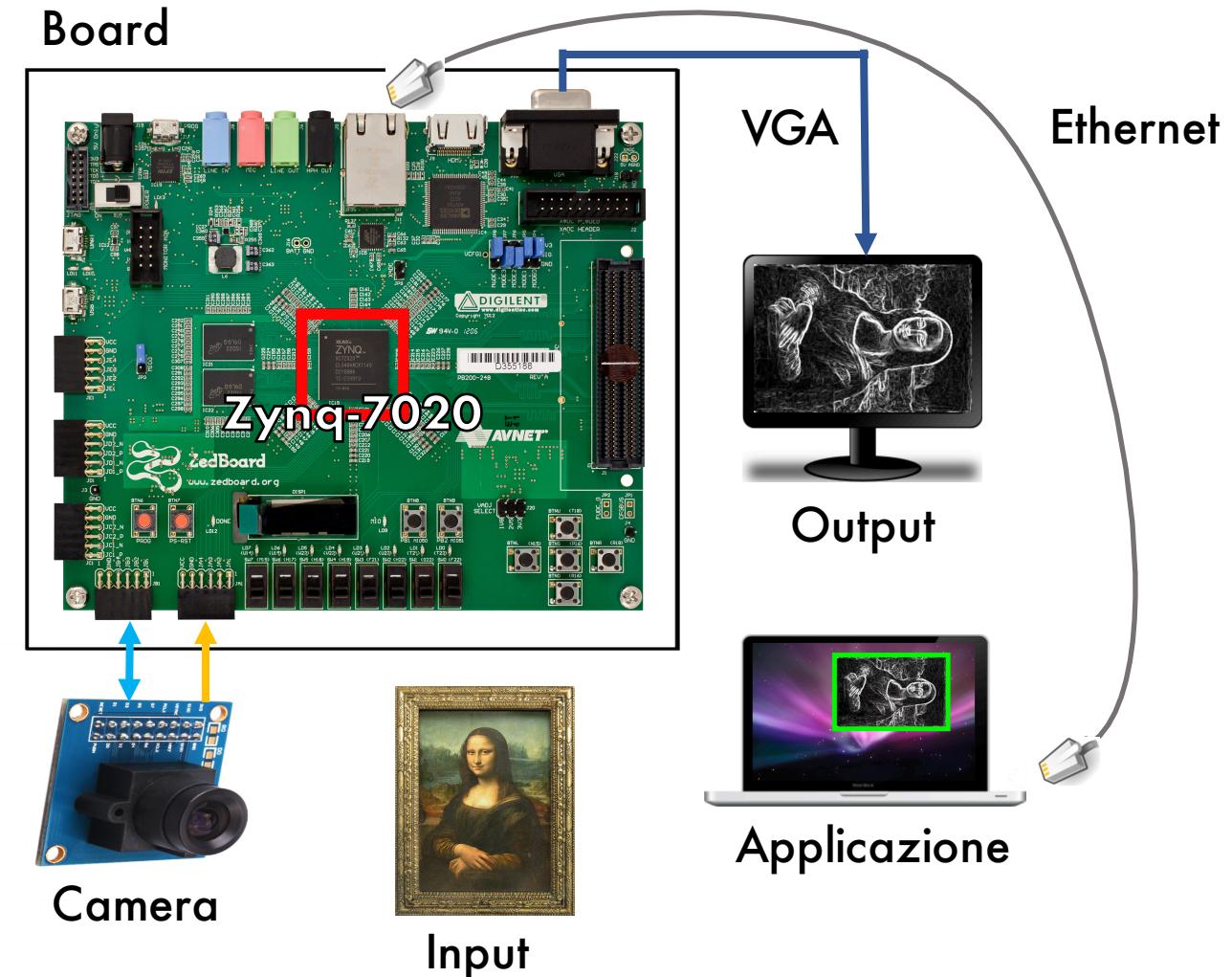
Candidato:
Marco Rossini

Contesto applicativo

Progettazione di filtro per elaborazione immagini

su piattaforma
↓

Zynq-7020 (FPGA + CPU)



Obiettivi

Implementazione su Zynq-7020 di algoritmo di convoluzione ottimizzato mediante utilizzo di:

- kernel separabili
- aritmetica fixed-point

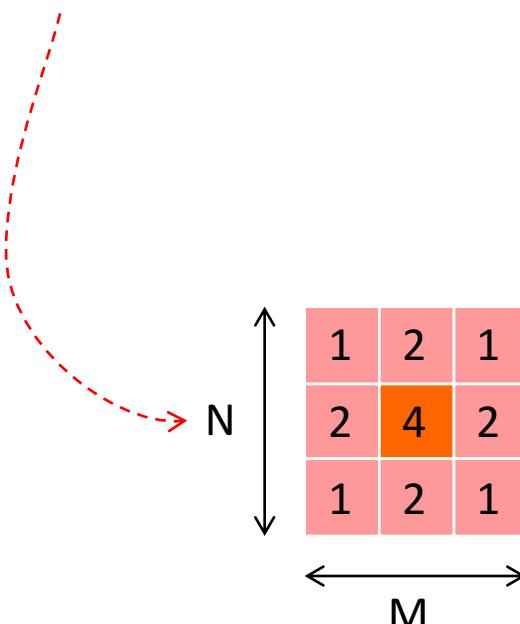
Strumento: **Vivado HLS IDE**



Il processo di convoluzione tradizionale

Elemento di filtraggio K :

Kernel



Operazione tra matrici:

$K \otimes I$

Elemento da filtrare I :

Immagine

Finestra di convoluzione

A 10x10 matrix labeled I representing the input image. A 3x3 submatrix is highlighted in red and orange, labeled "Finestra di convoluzione".

3	1	5	7	2	2	2	8	7	5	6	7
5	2	7	7	8	8	9	6	6	5	9	8
3	4	6	8	0	1	2	5	6	6	7	7
2	3	0	8	0	2	2	3	5	7	8	9
1	9	4	7	1	1	4	3	4	5	7	2
2	4	5	6	7	8	7	9	9	0	2	3

W

H

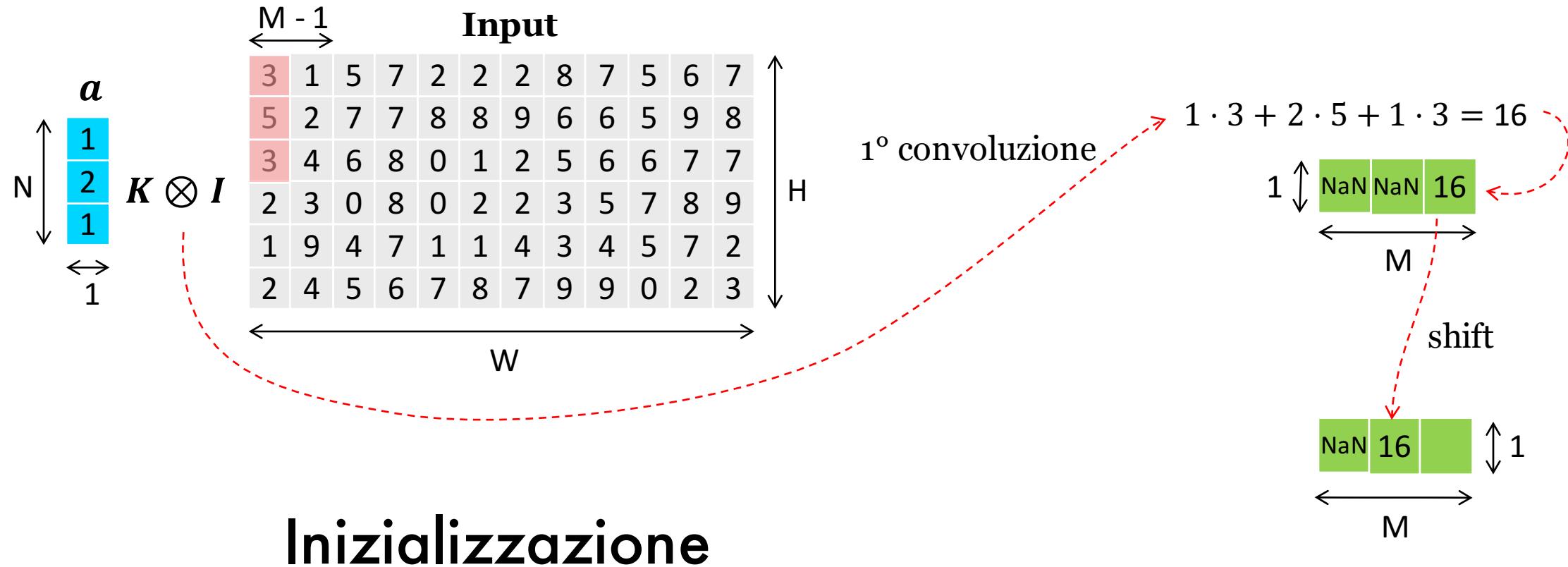
Kernel separabili

Ottenibili mediante *prodotto esterno* \otimes
tra un vettore colonna a e un vettore riga b

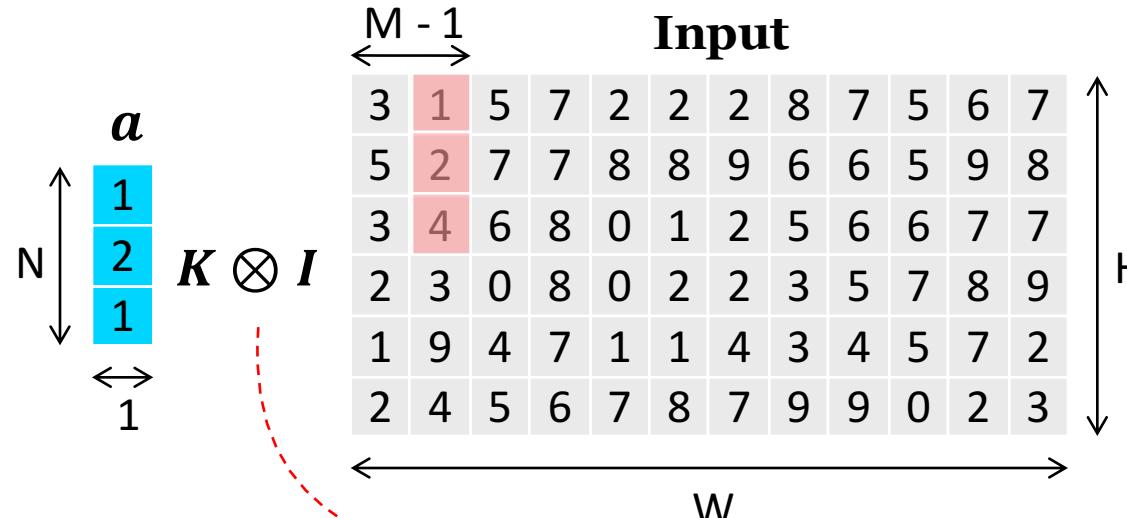
Ipotesi a fini
pratici

$$\begin{array}{c} \xrightarrow{\quad b \quad} \\ K = \begin{array}{|c|c|c|} \hline a_1 b_1 & a_1 b_2 & a_1 b_3 \\ \hline a_2 b_1 & a_2 b_2 & a_2 b_3 \\ \hline a_3 b_1 & a_3 b_2 & a_3 b_3 \\ \hline \end{array} \\ \xleftarrow{\quad a \quad} \end{array} = \begin{array}{|c|} \hline a_1 \\ \hline a_2 \\ \hline a_3 \\ \hline \end{array} \otimes \begin{array}{|c|c|c|} \hline b_1 & b_2 & b_3 \\ \hline \end{array} b$$

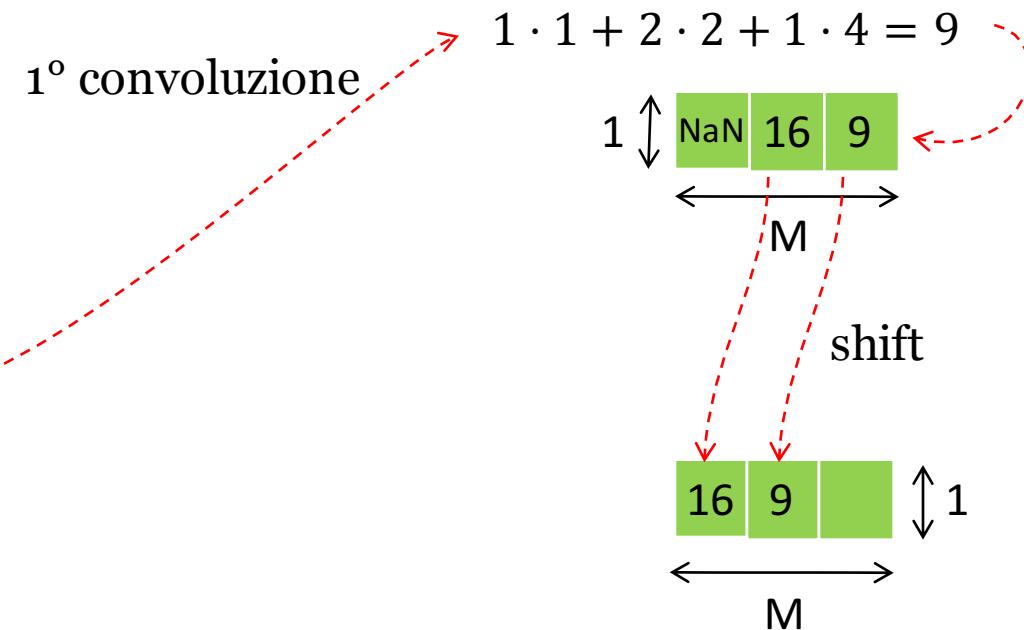
Il processo di convoluzione separabile



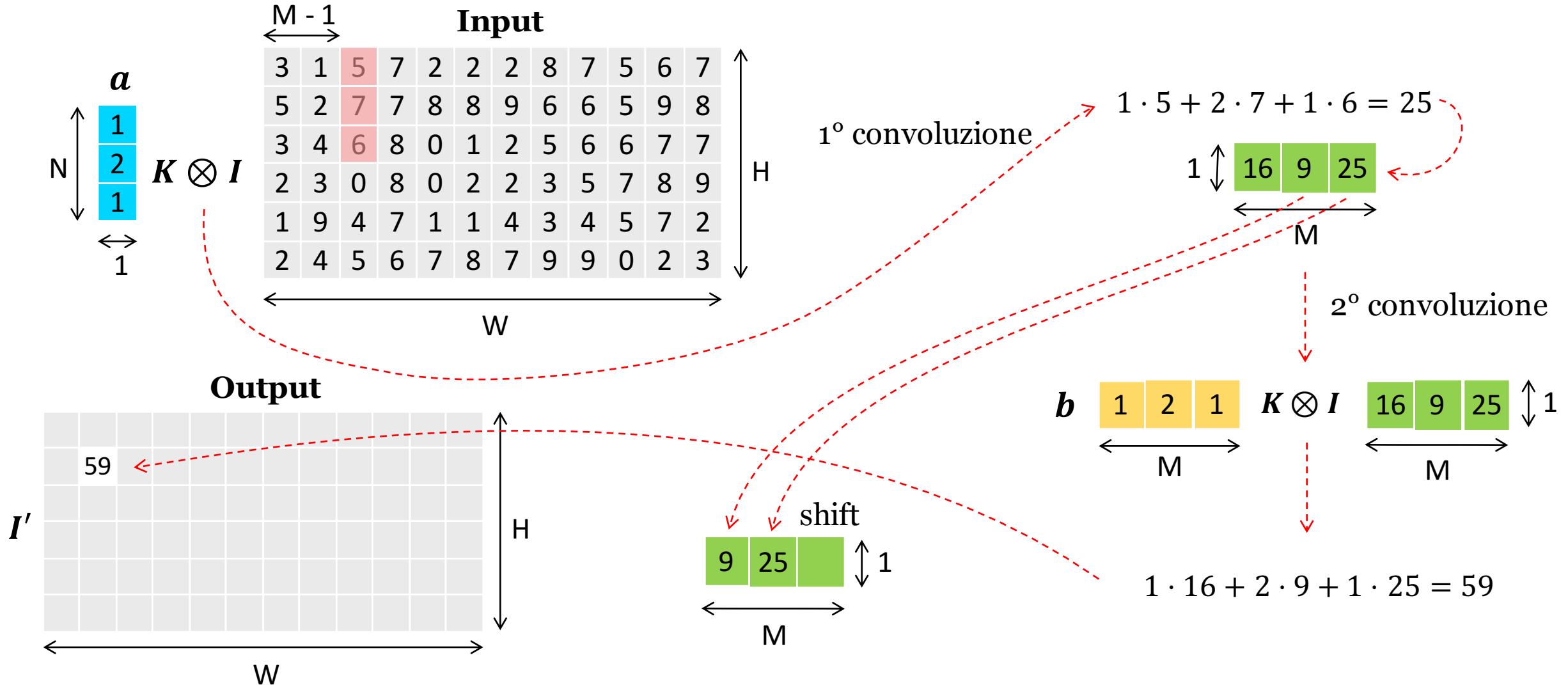
Il processo di convoluzione separabile



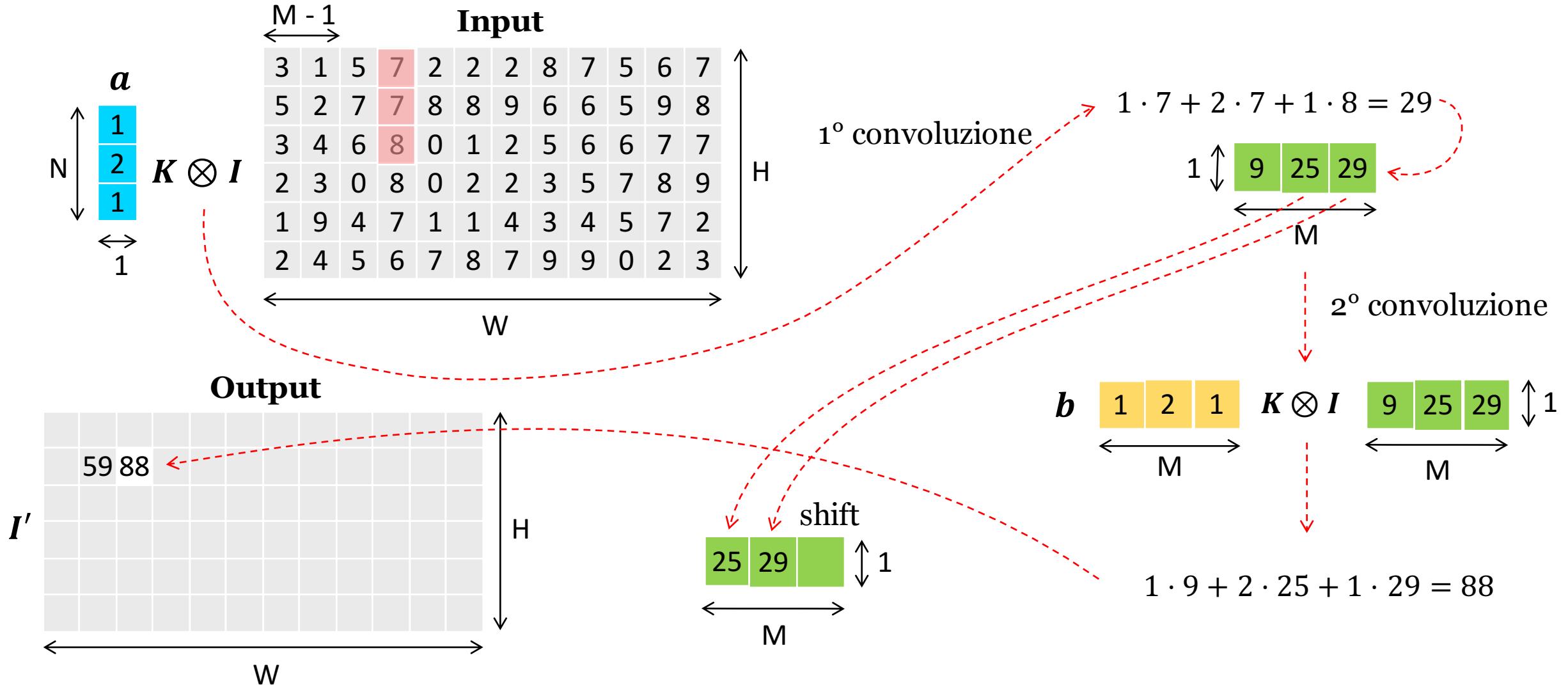
1° convoluzione



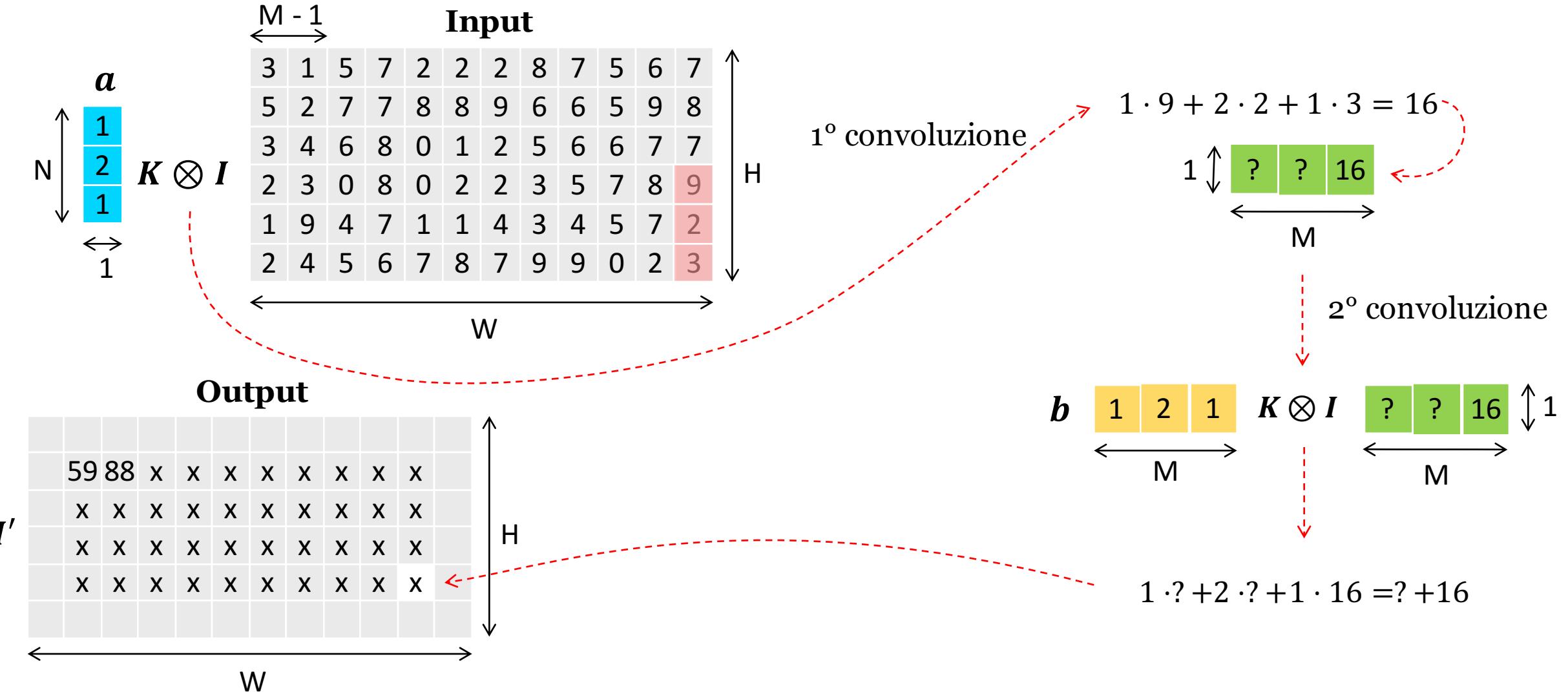
Il processo di convoluzione separabile



Il processo di convoluzione separabile



Il processo di convoluzione separabile



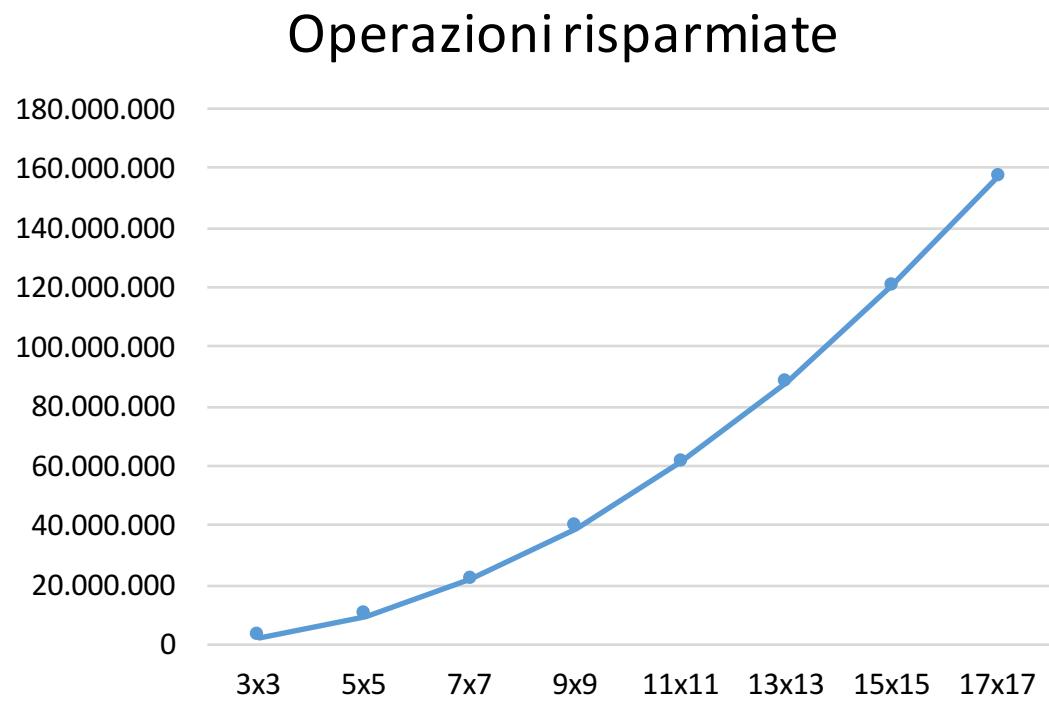
Complessità

N° operazioni algoritmo tradizionale = $2MN \cdot WH$

N° operazioni algoritmo separabile = $(M + 3N - 1) \cdot WH$

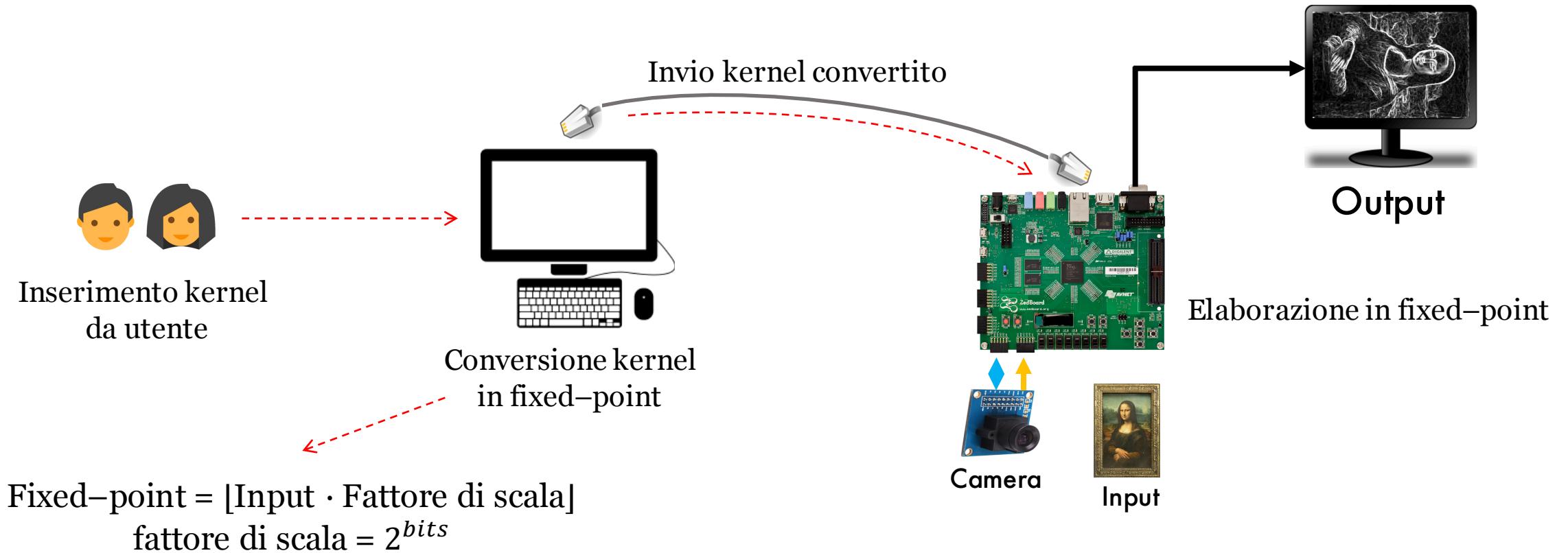
N° operazioni risparmiate = $(2MN - M - 3N + 1) \cdot WH$

Kernel	N° operazioni tradizionale	N° operazioni separabile	N° operazioni risparmiate
3x3	5 529 600	3 379 200	2 150 400
5x5	15 360 000	5 836 800	9 523 200
7x7	30 105 600	8 294 400	21 811 200
9x9	49 766 400	10 752 000	39 014 400
11x11	74 342 400	13 209 600	61 132 800
13x13	103 833 600	15 667 200	88 166 400
15x15	138 240 000	18 124 800	120 115 200
17x17	177 561 600	20 582 400	156 979 200

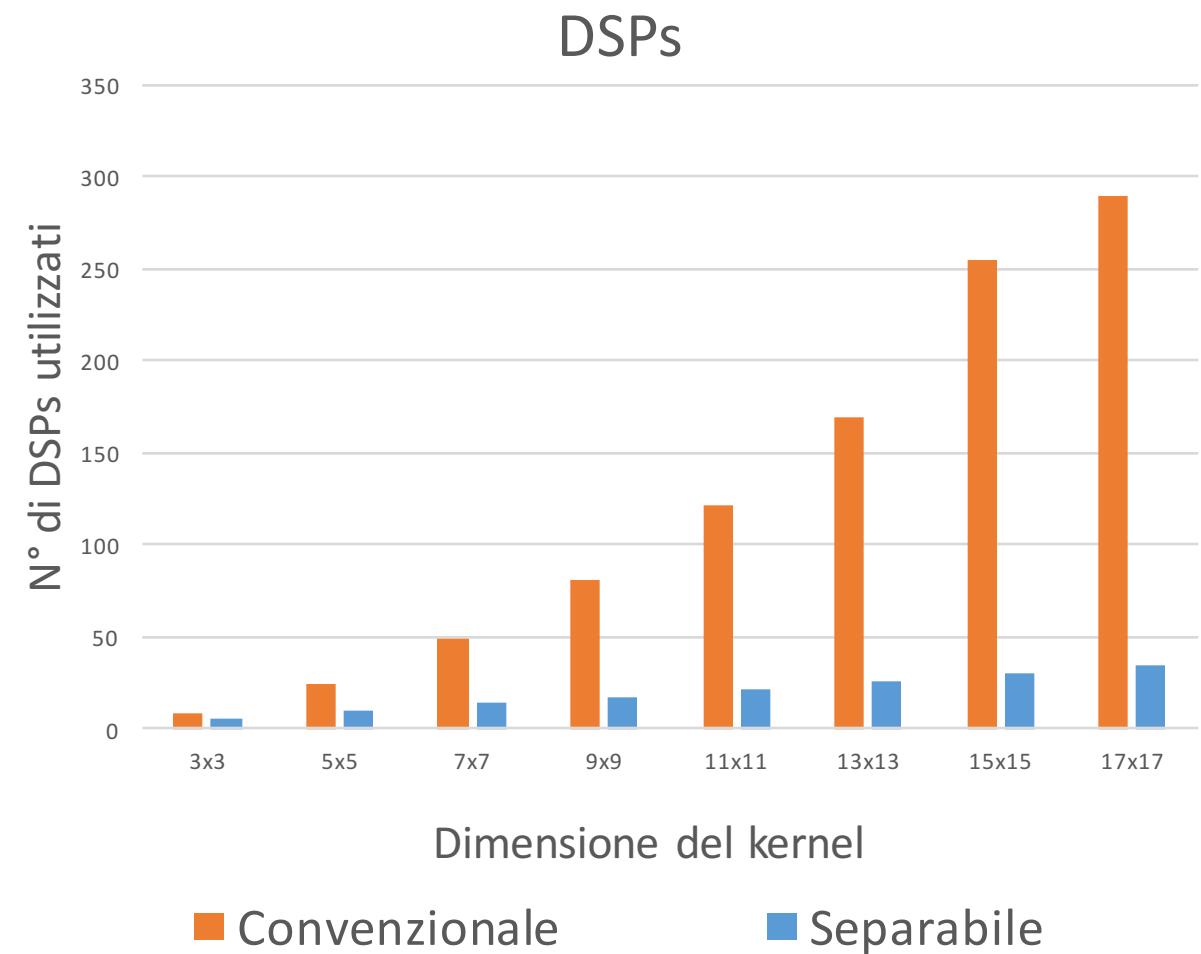
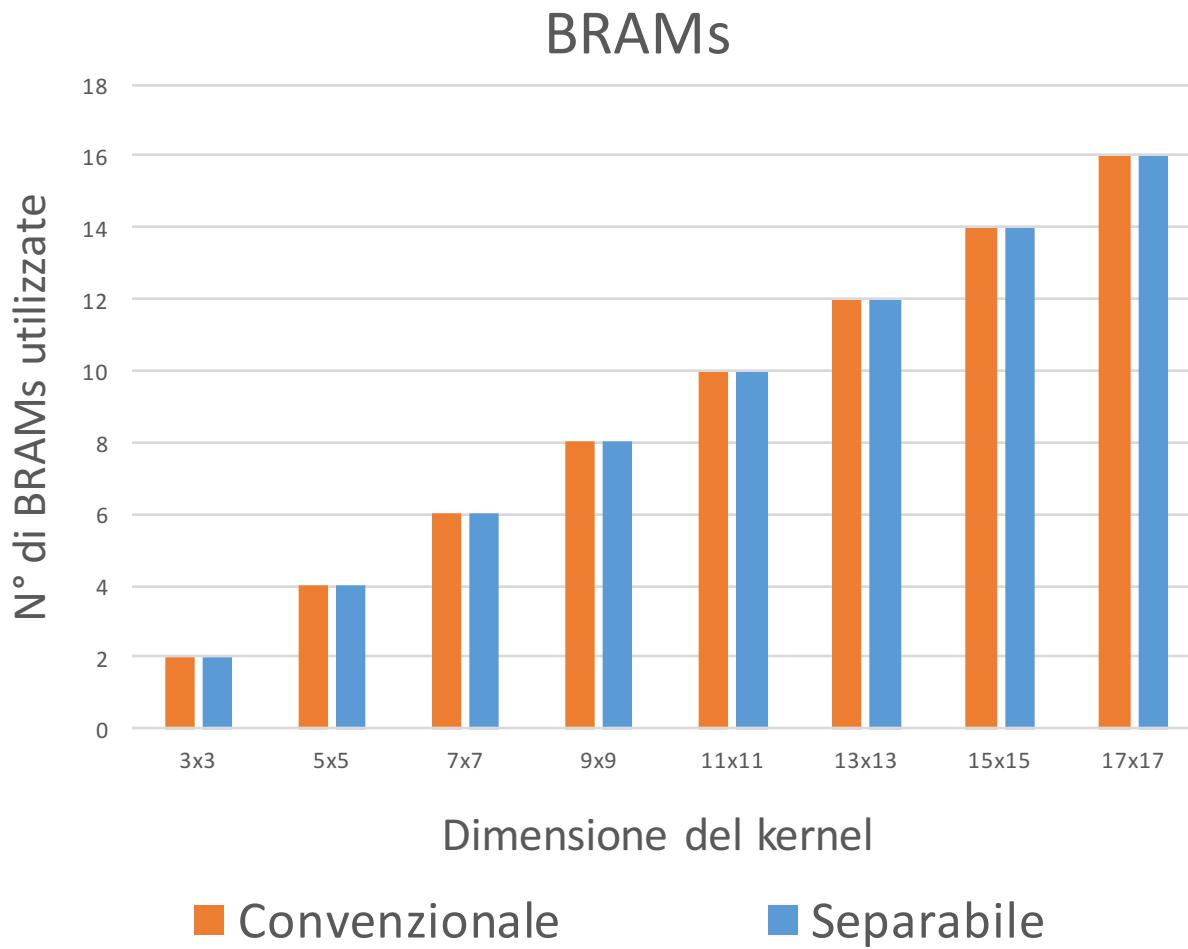


Fixed-point

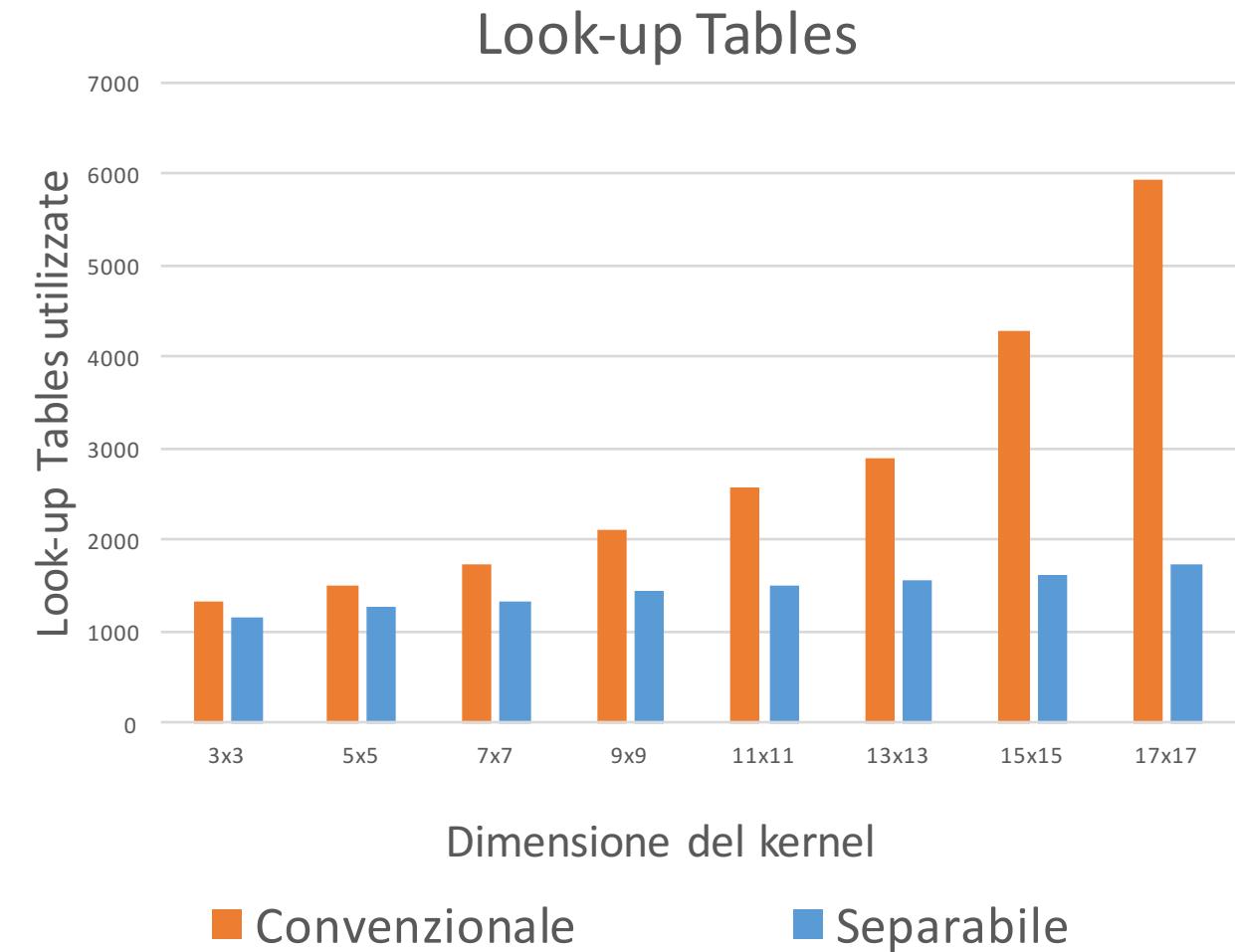
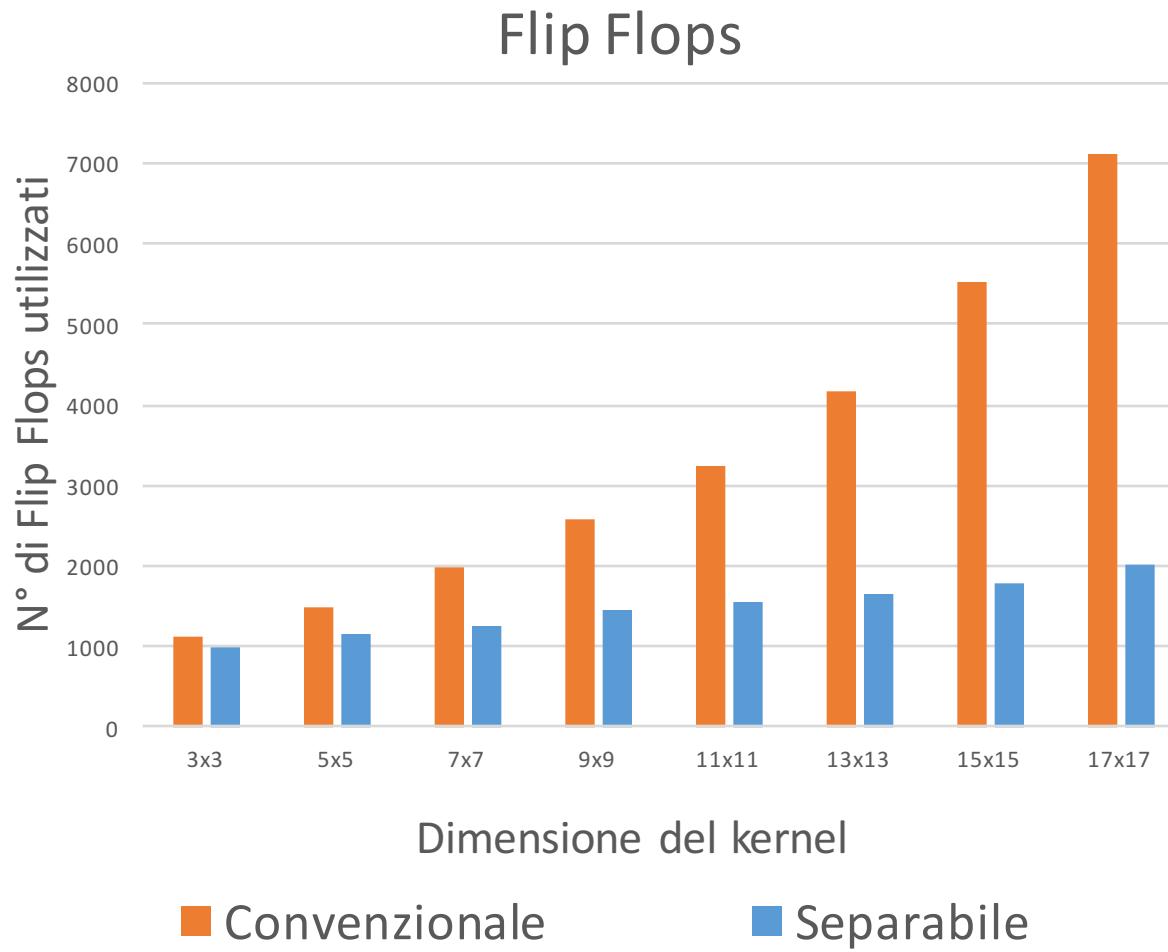
Vantaggi: efficienza di elaborazione



Risultati sperimentali



Risultati sperimentali



Conclusioni

- ✓ Ottimizzazioni forti per applicazioni basate su convoluzione
 - ✓ Fondamentali per porting di logiche complesse su dispositivi a risorse limitate (FPGA)

Sviluppi futuri:

porting di una CNN su FPGA sfruttando queste ottimizzazioni
(a seguito della definizione di un'opportuna architettura)

Grazie per l'attenzione