

# Report gruppo no. 13

## Scelte implementative

Abbiamo scelto di implementare i canali come prima dimensione dell'array 3d, poi righe e colonne, perché concettualmente ci risultava più facile pensare la matrice in questo modo, in quanto aveva senso accedere prima al canale e poi alla matrice del canale, quindi  $k \times h \times w$ .

## Funzioni aggiuntive rispetto a ip\_lib.h

```
/* funzione invocata da clamp, Comprime un valore all'interno del range [lo,hi]*/
```

```
float clamp_f(float f, float lo, float hi)
{
    if (f > hi)
    {
        return hi;
    }
    else if (f < lo)
    {
        return lo;
    }
    else
    {
        return f;
    }
}
```

```
/*Ritorna una matrice ad una dimensione i cui valori sono la media delle k dimensioni della matrice originale*/
```

```
/*Utilizzato nella prima versione del progetto per risolvere il problema di avere filtro ed immagine con un numero di canali diversi all'interno di ip_mat_convolve, è stata successivamente tolta in seguito alle modifiche della versione 1.1.1 del progetto*/
```

```
ip_mat* ip_mat_1D_average(ip_mat* t)
{
```

```

unsigned int i, j, k;
ip_mat* avg;

avg = ip_mat_create(t->h, t->w, 1, 0);

for (i = 0; i < t->h; i++)
{
    for (j = 0; j < t->w; j++)
    {
        float valSum = 0;
        for (k = 0; k < t->k; k++)
        {
            valSum += t->data[k][i][j];
        }

        /*Media delle k dimensioni*/
        avg->data[0][i][j] = valSum / t->k;
    }
}
compute_stats(avg);
return avg;
}

```