Entiol Liko: Lab 5

enliko@student.ethz.ch

1 Naive Implementation

This implemenation can be seen as the fast batch implementation but with batch size equal to 1.The implemented logic is the same in both cases.

1.1 Distance Gaussian

I calculated the squared distance between the pixel and all the other pixels by extending the dim of x by 1 with size equal to the total number of pixels. Thanks to this new matrix I can just substract the two matrixes $x\_$ and X and then squaring, at the end I just sum over the dimension 1. I can use this new matrix to calculate the gaussian distance.

1.2 Update Point

I can calculate the shift for this specific pixel using the formula, where in the numerator I use matmult to calculate the multiplication between the distance and X and in the denominator I just sum over the distance vector.

2 Fast Implementation

The fast implementation is very similar the native implementation. Here we use batches of size 256 insead of a single pixel.The only important change here is the fact that we use the function cdist to calculate the distance between the batch and the matrix X and the fact that we are more careful when with the dimensions in the function $\text{update}_{p}oint_{b}atch$.

The results are the following:
Fast Implementation: 4.71 sec Slow Implementation: 14.66 sec Image Result: