ETH Zürich
Mathematical Foundations of
Computer Graphics and Vision
Spring 2019

M. Arda Düzceker
18-950-485
ardad@student.ethz.ch

EXERCISE 3

MLS FOR CURVES, MESHES AND IMAGES

To run the codes of this exercise, *../code/* must be the working directory. Then, running corresponding parts performs all the necessary calculations required and generates the plots.

1. CURVE AND SURFACE RECONSTRUCTION USING MLS

1.1. Curves Derivation

The solution of moving least squares problem is acquired by solving the general form,

$$\underset{\boldsymbol{c}}{\operatorname{argmin}} E(\boldsymbol{c}) = \sum_i \phi_i(x)\ (\boldsymbol{b}(x_i)^T \boldsymbol{c} - f_i)^2$$

When locally fit polynomial, $f(x) = \boldsymbol{b}(x)^T \boldsymbol{c}$, consists of a single constant term, then $\boldsymbol{b} = 1$ and $\boldsymbol{c} = c_0$ which results in $f(x) = c_0$. Therefore, the problem definition can be rewritten as,

$$\underset{c_o}{\operatorname{argmin}} E(c_0) = \sum_i \phi_i(x)\ (c_0 - f_i)^2$$

To find the value of c_0 which minimizes the error function, we can take the derivative of the error with respect to c_0 and equate to 0. Note that the result is a simple weighted average in our case.

$$\frac{d}{dc_0} E(c_0) = \frac{d}{dc_0} \sum_i \phi_i(x)\ (c_0 - f_i)^2 := 0$$

$$2 \sum_i \phi_i(x)(c_0 - f_i) = 0$$

$$\sum_i \phi_i(x) c_0 = \sum_i \phi_i(x) f_i$$

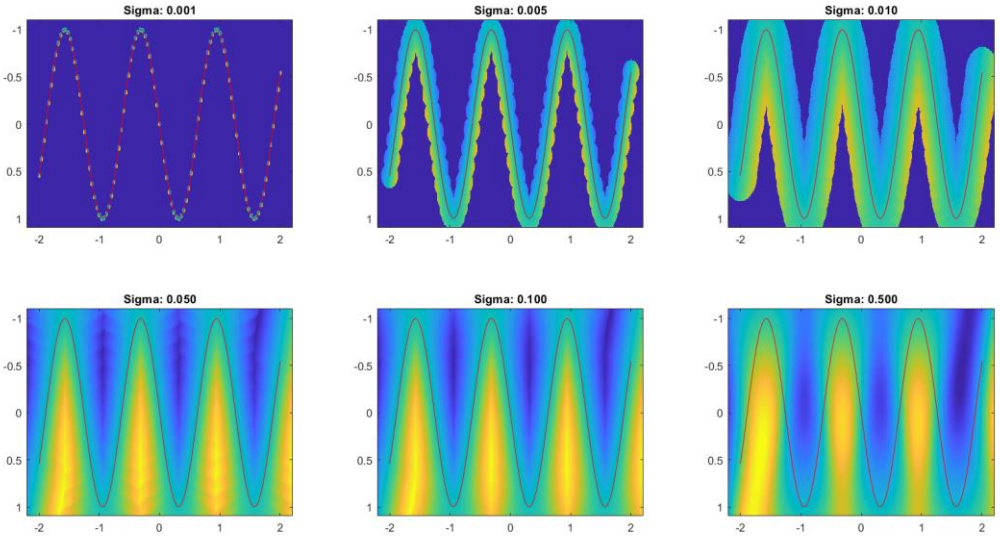
$$c_0 = \frac{\sum_i \phi_i(x) f_i}{\sum_i \phi_i(x)}$$

M. Arda Düzceker
18-950-485
ardad@student.ethz.ch

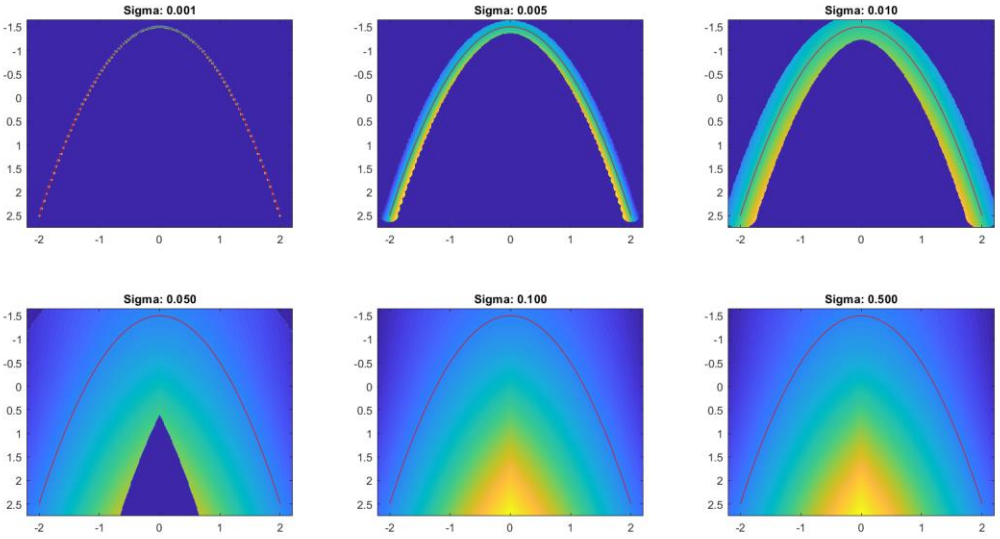
1.2. Curves plotting

MLS for curve fitting algorithm, utilizing the $f(x)$ found in 1.1 is tested with different values of sigma for Gaussian weighting function. The grid resolution is always kept as 0.01 in both axes. According to the results, a balanced performance between smoothness and continuity is achieved with sigma = 0.005 in all three cases. Running *part1_2.m* is enough to get the results.

Curve Data 1:

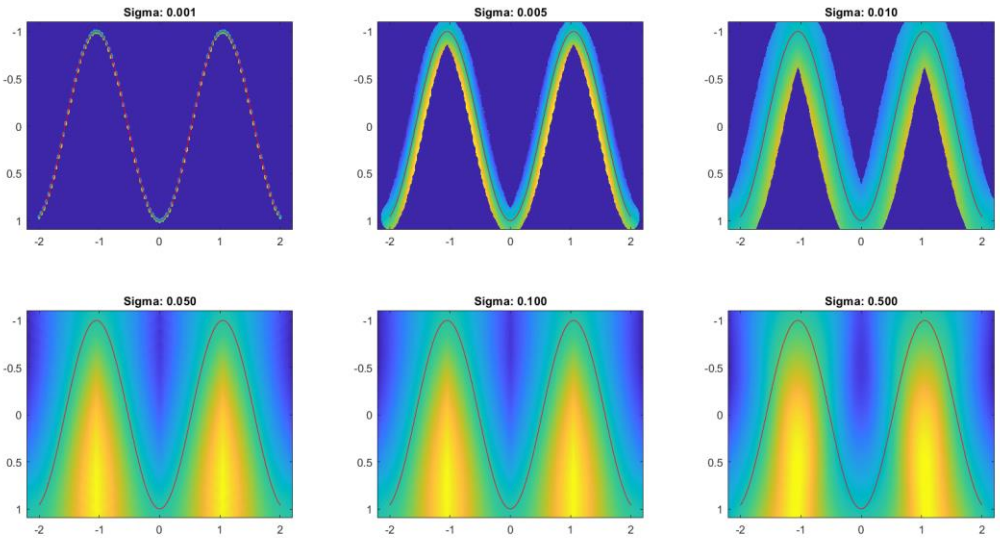


Curve Data 2:



M. Arda Düzceker
18-950-485
ardad@student.ethz.ch

Curve Data 3:



1.3. Smoothing Meshes

The derivation of the gradient function is as follows:

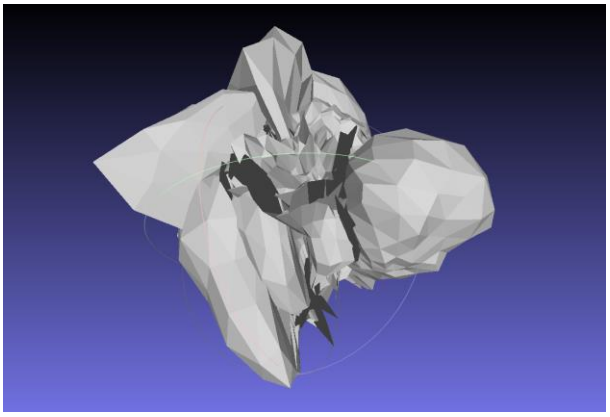
$$\nabla f(x) = \nabla \frac{\sum_i \phi_i(\boldsymbol{x}) f_i}{\sum_i \phi_i(\boldsymbol{x})} = \nabla \frac{\sum_i \phi_i(\boldsymbol{x}) \boldsymbol{n}_i^T (\boldsymbol{x} - \boldsymbol{x}_i)}{\sum_i \phi_i(\boldsymbol{x})}$$

$$= \frac{\nabla\left(\sum_i \phi_i(\boldsymbol{x}) \boldsymbol{n}_i^T (\boldsymbol{x} - \boldsymbol{x}_i)\right)\left(\sum_i \phi_i(\boldsymbol{x})\right) - \left(\sum_i \phi_i(\boldsymbol{x}) \boldsymbol{n}_i^T (\boldsymbol{x} - \boldsymbol{x}_i)\right)\nabla\left(\sum_i \phi_i(\boldsymbol{x})\right)}{\left(\sum_i \phi_i(\boldsymbol{x})\right)^2}$$

$$= \frac{\sum_i \left(\nabla \phi_i(\boldsymbol{x})\right) \boldsymbol{n}_i^T \boldsymbol{x} + \sum_i \phi_i(\boldsymbol{x}) \boldsymbol{n}_i^T - \sum_i \left(\nabla \phi_i(\boldsymbol{x})\right) \boldsymbol{n}_i^T \boldsymbol{x}_i - \frac{\sum_i \phi_i(\boldsymbol{x}) \boldsymbol{n}_i^T (\boldsymbol{x} - \boldsymbol{x}_i)}{\sum_i \phi_i(\boldsymbol{x})} \left(\sum_i \nabla \phi_i(\boldsymbol{x})\right)}{\sum_i \phi_i(\boldsymbol{x})}$$

$$\nabla f(\boldsymbol{x}) = \frac{\sum_i \phi_i(\boldsymbol{x}) \boldsymbol{n}_i^T + \sum_i \left(\nabla \phi_i(\boldsymbol{x})\right) \left(\boldsymbol{n}_i^T (\boldsymbol{x} - \boldsymbol{x}_i) - f(\boldsymbol{x})\right)}{\sum_i \phi_i(\boldsymbol{x})}$$

However, I did not manage to implement it correctly. In the code folder, part1_3.m contains all the necessary computations but the result *trial.off* which is acquired after only one iteration of smoothing *bunny.off* is not correct.

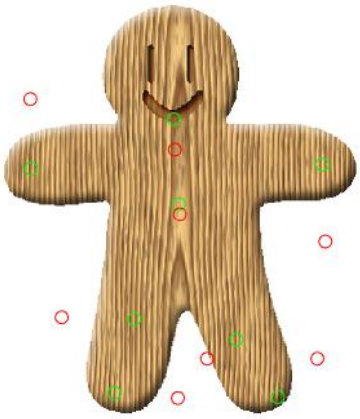


M. Arda Düzceker
18-950-485
ardad@student.ethz.ch

2. IMAGE DEFORMATION USING MOVING LEAST SQUARES

The algorithm is implemented according to the provided paper. Running *part2.m* performs necessary calculations and outputs the images.

For warping, reverse mapping is used to avoid artifacts (black spots in the warped image).



M. Arda Düzceker
18-950-485
ardad@student.ethz.ch

Affine Warped Image

Similarity Warped Image

Rigid Warped Image