The aim of this project is to explore and develop fast and stable algorithms, standard reference algorithms, and measurement procedures for non-linear geometrical Gaussian and spline filtration. Focusing on:

- Exploration of suitable numeric models for robustness and stability of non-linear filters.
- Creation of fast algorithms for the efficient implementation of non-linear Gaussian and spline filters.
- Development of standard reference algorithms implementing the definitions according to the ISO 16610 series of standards.

\[
\sum_{k=0}^{n-1} \rho\left(z(x_k) - r(x_k)\right) + \lambda \int _{x_0}^{x_1} \frac{d^2 s(x)}{dx^2}^2 \, dx \to \operatorname{Min} \ s(x_k)
\]

**Generalised Spline filter**

<table>
<thead>
<tr>
<th>Filter Type</th>
<th>Weighted Linear</th>
<th>Nonlinear</th>
<th>Robust</th>
</tr>
</thead>
<tbody>
<tr>
<td>L2</td>
<td>$\rho(z, \beta) = (z - \beta)^2$</td>
<td>$\rho(z, \beta) =</td>
<td>z - \beta</td>
</tr>
<tr>
<td>L1</td>
<td>$\rho(z, \beta) = (z - \beta)$</td>
<td>$\rho(z, \beta) =</td>
<td>z - \beta</td>
</tr>
<tr>
<td>Huber</td>
<td>$\rho(z, \beta) = \sqrt{c^2 +</td>
<td>z - \beta</td>
<td>^2}$</td>
</tr>
<tr>
<td>Cauchy</td>
<td>$\rho(z, \beta) = \frac{1}{2c^2} (c^2 +</td>
<td>z - \beta</td>
<td>^2)$</td>
</tr>
<tr>
<td>Tukey</td>
<td>$\rho(z, \beta) = \frac{1}{2c^2} (c^2 +</td>
<td>z - \beta</td>
<td>^2)$</td>
</tr>
</tbody>
</table>

\[
\int_0^1 \rho\left(z(\xi) - w(x) - \beta_1(x)(\xi - x) - \beta_2(x)(\xi - x)^2\right) s(\xi - x) \, d\xi \\
\to \operatorname{Min} \ w(x), \beta_1(x), \beta_2(x)
\]

**Generalised higher order gaussian regression filter for 2D Profile**

\[
\int_0^1 \rho\left(z(\xi, \eta) - w(x, y) - \beta_{10}(x, y)(\xi - x) - \beta_{11}(x, y)(\eta - y) - \beta_{20}(x, y)(\xi - x)^2 - w(x, y)\right) s(\xi - x, \eta - y) \, d\xi \, d\eta \\
\to \operatorname{Min} \ w(x, y), \beta_{10}(x, y), \beta_{11}(x, y), \beta_{20}(x, y), \beta_{11}(x, y), \beta_{22}(x, y)
\]

**Fast algorithms:**
1. Convolution to FFT;
2. Pre-calculation;
3. Separable in rows and cols

**Significant speed improvement:**
For a typical 60,000 pts data, 100 ms is needed compared with traditional algorithms need a few hours