GAMMA Technical Report: Interpolation and resampling

Christophe Magnard, Charles Werner, and Urs Wegmüller

Gamma Remote Sensing, Worbstrasse 225, CH-3073 Gümligen, Switzerland

GAMMA REMOTE SENSING

Nov 2017

TABLE OF CONTENTS

SUMMARY AND RECOMMENDATIONS

Interpolation is a crucial operation in the processing of SAR data. The quality of an interpolation method is characterized by the ability to preserve the original signal when performing the interpolation step. Depending on the data, insufficient quality interpolation may result in a smoothing of the data (low-pass filter effect), create ringing artifacts, decrease the average intensity, and alter the phase. Preserving the original signal is particularly important for complex-valued data in interferometry; there, insufficient quality interpolation will result in higher interferometric phase noise / lower coherence.

The interpolation methods in the Gamma software were reviewed and updated. Very high-quality B-spline and Lanczos interpolation methods have been implemented. They replace the previously-used truncated sinc interpolation (Lanczos is a variant of sinc interpolation). An option is available for setting the size of the B-spline or Lanczos interpolation kernel; it permits adapting the interpolation based on the data type and the desired processing speed. Orders ranging from 2 to 9 are available for both interpolation methods, the B-spline kernel size is equal to the *order + 1* while the Lanczos kernel size is equal to *2 · order*. The interpolation methods were tested in different scenarios; these are reported in the next sections. Tab. 1 summarizes the recommended interpolation method depending on the input data.

Single look complex (SLC)	In most cases, use a mid to high order (4 to 9) Lanczos interpolation. For oversampled data, a mid to high order (4 to 9) B-spline interpolation gives the best results.
Multi-look intensity data (MLI)	The interpolation should be performed on the square root of the data. A mid-order (3 to 5) B-spline interpolation is recommended.
Digital elevation model (DEM) Non-periodic / undersampled real-valued image	Use a bicubic spline or low to mid order (2 to 5) B-spline interpolation.
Image with indexed color scale (e.g. 8-bit TIFF image)	Only the nearest neighbor interpolation is available.
In case of ringing effects	Use a lower order Lanczos / B-spline interpolation or the bicubic spline. Perform the interpolation on the square root of the data (real-valued data).

Tab. 1 Recommended interpolation method depending on the input data

Lanczos and B-spline interpolation methods both provide very high-quality interpolation for single look complex (SLC) data. In both cases, the higher the order, the better the interpolation for SLC data, at the cost of longer processing time. The differences between both methods are summarized in Tab. 2.

In the case of multi-look intensity (MLI) data, the Lanczos interpolation tends to introduce a slight bias in the data. This effect decreases for larger order interpolation kernels, and is probably caused by the lack of the partition of unity property: the sum of the interpolation kernel coefficients is not exactly equal to 1, i.e. the interpolation of a constant signal will not be constant. The B-spline interpolation does not suffer from this issue and is therefore recommended. Results are otherwise very similar for a given interpolation kernel order. The interpolation should always be performed on the square root of the intensity (i.e. on the amplitude).

Note that the bicubic spline interpolation on the log of the data is still available in the programs for continuity, but is not recommended.

Lanczos interpolation	B-spline interpolation
Does not have the partition of unity property. Particularly problematic for low order functions: order 2 should not be used at all, order 3 is not recommended.	Has the partition of unity property, no issue using low order B-spline interpolation.
For a given interpolation order, slightly better results when the data is sampled close to Nyquist rate: the amplitude / intensity is slightly better preserved.	For a given interpolation order, slightly better results when the data is oversampled.

Tab. 2 Lanczos vs. B-spline interpolation of complex-valued data.

1 INTERPOLATION THEORY

Interpolation consists of generating a continuous function from an array of discrete values. Interpolation methods can be split into two categories: classical and generalized interpolation methods [1]. The 2D case of the classical interpolation method is shown in equation (1).

$$f(\vec{x}) = \sum_{\vec{k}\in\mathbb{Z}^2} f_{\vec{k}} \cdot \varphi_{\mathrm{int}}(\vec{x}-\vec{k}), \forall \vec{x} = (x_1, x_2) \in \mathbb{R}^2 \tag{1}$$

where $f(\vec{x})$ is the interpolated continuous image, $f_{\vec{k}}$ the discrete image, $\varphi_{\mathrm{int}}(\vec{x})$ the interpolating function such as nearest neighbor, linear, cubic spline or truncated sinc *interpolant* such as the Lanczos interpolant.

Generalized interpolation uses pre-computed discrete coefficients instead of the discrete image:

$$f(\vec{x}) = \sum_{\vec{k}\in\mathbb{Z}^2} c_{\vec{k}} \cdot \varphi(\vec{x}-\vec{k}), \forall \vec{x} = (x_1, x_2) \in \mathbb{R}^2 \tag{2}$$

with $\varphi(\vec{x})$ a *non-interpolant* function and where the coefficients $c_{\vec{k}}$ are calculated by solving the linear system of equations defined by:

$$f_{\vec{k}_0} = \sum_{\vec{k}\in\mathbb{Z}^2} c_{\vec{k}} \cdot p_{\vec{k}_0-\vec{k}}, \forall \vec{k}_0 \in \mathbb{Z}^2 \tag{3}$$

with $p_{\vec{k}} = \varphi(\vec{k})$.

Note that in practice, the number of samples $\vec{k}_0$ is finite. The B-spline interpolation ([2], [3]) uses the generalized interpolation method.

The partition of unity or reproduction of the constant is an important property for interpolation. It involves that the sum of the kernel coefficients used to interpolate a value is always equal to 1 (see equation (4)). See [1] for additional information.

$$\sum_{\vec{k}\in\mathbb{Z}^2} \varphi(\vec{x}-\vec{k}) = 1, \forall \vec{x} = (x_1, x_2) \in \mathbb{R}^2 \tag{4}$$

In signal processing, from Nyquist – Shannon theorem, the sinc interpolant is typically considered as the ideal interpolant for bandlimited signals. The sinc function has however an infinite size and thus is not usable for finite signals. More generally, the ideal interpolation method is characterized by the ability to preserve the original signal when performing the interpolation step.

The performance of an interpolation kernel may vary depending on the data: an interpolation kernel closely matching a sinc interpolation may well be ideal for interpolating a periodic / bandlimited signal, it may create unwanted ringing artifacts in the case of a non-periodic / non-bandlimited signal. In addition, the practical selection of an interpolation method will also depend on a trade-off between quality and processing time.

2 INTERPOLATION METHODS AND PROPERTIES

Several interpolation methods are implemented in the Gamma software:

Nearest neighbor

The nearest neighbor interpolation gives the value of the nearest discrete point to the interpolated value. It is the fastest interpolation method but yields poor results for usual signals. It may be the only available method for data using non-linear scales, such as images using indexed values.

Cubic spline

The cubic spline (Catmull-Rom or Keys) is a rough approximation of the sinc function. Its kernel is defined between -2 and 2 and equals to 0 beyond.

The implemented cubic-spline interpolation has the partition of unity property, and its derivative is defined and continuous everywhere.

B-spline

The B-spline interpolation is a generalized interpolation method. It results in an approximation of the sinc spanning the whole input data length. The sinc approximation is more accurate for higher-order B-splines. B-splines orders from 2 to 9 are implemented in the Gamma software; they use kernels spanning from 3 to 10 cells.

B-spline interpolation requires the computation of the array of coefficients prior to the effective interpolation, which may be inefficient in some cases, such as when only few values need to be interpolated. It might also result in poor results for small arrays.

Neglecting the coefficient calculation, the B-spline interpolation has a very high quality vs. processing time ratio. The resulting efficiency thus depends on the proportion of processing time taken for the coefficient calculation.

B-spline interpolation has the partition of unity property, the interpolated signal and its derivative are defined and continuous everywhere.

Truncated sinc

The truncated sinc is a sinc function defined over a finite size. The values beyond the kernel size are set to 0. The truncated sinc causes major ripples in frequency domain and does not have the partition of unity property (i.e. the interpolation of a constant signal will not be constant). A window has to be applied on the truncated sinc to minimize these issues. Many windows have been proposed with various advantages and disadvantages; the Lanczos interpolation described below uses the main lobe of a sinc as its window.

A few programs in the Gamma software may still use a truncated sinc multiplied by a Kaiser window. These programs will be investigated and updated in the future.

Lanczos interpolation

The Lanczos interpolation consists of the multiplication of a truncated sinc signal by the main lobe of another sinc signal stretched to the window size [4]. The size of the interpolation kernel can be easily varied; high order Lanczos interpolation yields very good results in SAR applications. Lanczos orders from 2 to 9 are implemented in the Gamma software; they use kernels spanning from 4 to 18 cells.

The main drawback of the Lanczos interpolation is the processing time due to the computation time required by the *sine* function. To overcome this issue, a look-up table containing a discretized Lanczos function is typically precomputed and used for the interpolation.

The interpolated signal is continuous everywhere, and its derivative is defined and continuous everywhere. Lanczos interpolation does not have the partition of unity property, although it is very close when using larger kernels.

FFT-based resampling

The sinc function corresponds to a rect function in frequency-domain. Zero-padding data in frequency domain thus corresponds to a sinc interpolation in time-domain. This method yields high quality results for mid-size to large arrays. However, it is relatively slow for large arrays and yields poor results in case of very small arrays.

Another method for achieving FFT-based resampling is by zero-interleaving the time-domain data; the array is then transformed into frequency domain, and a low-pass filter is applied to remove the artificial spectrum repetitions introduced by the zero-interleaving.

One drawback of FFT-based interpolation is that contrary to conventional interpolation, no continuous image is built. We would have to repeat the full process for each interpolated value, which would be extremely inefficient, and only rational values $\vec{x} = (x_1, x_2) \in \mathbb{Q}^2$ could be interpolated.

2.1 Data up-sampling

Data up-sampling by an integer factor can be achieved using FFT-based resampling.

It can also be achieved using the other interpolation methods in a more efficient way than for general interpolation at "random" locations: the position of the interpolated points relative to the original points is constant over the whole dataset. This allows separating the 2D interpolation into two 1D interpolations. Moreover, the original values do not need to be recomputed. This reduces the number of computations by

$$\frac{m \cdot ovr}{3 \cdot (ovr - 1)} \tag{5}$$

where m is the size of the interpolation kernel and ovr the oversampling factor. For example, with $ovr = 2$ and $m = 12$, the number of computations is divided by 8. In addition, the kernel can be precomputed without approximation error, thus further accelerating the interpolation process.

3 MODIFIED PROGRAMS

The modified programs (per December 2017 upgrade) are listed in Tab. 3, together with the available interpolation method(s).

In addition, several programs in the Gamma software use interpolation for internal computations (they do not produce an image filled with interpolated values). They include the offset estimation programs, where temporary resampling and sub-pixel estimation of the maximum of the correlation function require interpolation methods: Lanczos, B-spline and FFT-based resampling are used depending on the function and program.

The updated programs are *gc_map2* (program added in the mid-2017 release), *offset_pwr*, *offset_pwr_tracking*, *offset_pwr_tracking2*, *offset_pwr_list*, *offset_pwrm*, *offset_pwr_trackingm*, *offset_pwr_trackingm2*, and *xpt_slc*.

	Nearest-neighbor	**Bicubic spline**	**B-spline**	**Lanczos**	**Sqrt**
dem_trans	Available	Available	Available order 2-5	-	-
geocode_back	Available	Available	Available order 2-9	Available order 2-9	Available
interp_data	Available	Available	Available order 2-9	Available order 2-9	Available
map_trans	Available	Available	Available order 2-9	Available order 2-9	Available
MLI_interp_lt	-	-	Always used order 2-9	-	Always used
resamp_image	Available	Available	Available order 2-9	Available order 2-9	Available
rotate_image	Available	Available	Available order 2-9	Available order 2-9	Available
SLC_interp	-	-	Available order 4-9	Available order 4-9	-
SLC_interp_lt	-	-	Available order 4-9	Available order 4-9	-
SLC_interp_map	-	-	Available order 4-9	Available order 4-9	-
SLC_interp_S1_TOPS	-	-	Available order 4-9	Available order 4-9	-
SLC_interp_lt_S1_TOPS	-	-	Available order 4-9	Available order 4-9	-

Tab. 3 List of modified programs and available interpolation method(s).

4 VALIDATION

This section describes the tests that were performed to compare and validate the interpolation methods depending on the input data and the obtained results.

4.1 Rotation tests

Rotation tests were performed on SLC and MLI data to investigate how well the original data are preserved depending on the interpolation method. In both cases, two tests were performed: one with the risk of having undersampled data, and one on oversampled data to avoid this risk.

4.1.1 Complex-valued SAR data

In the first experiment, a TerraSAR-X SLC was directly rotated 45° and rotated back, using all available interpolation methods and orders. The results were compared to the original data by computing several statistical values. Note that the effect of the two interpolations is measured with this experiment!

The most important results are reported in the following figures. The slope of the amplitude regression in Fig. 2 informs about how well the amplitude (and intensity) is preserved depending on the target intensity. The median of the amplitude difference in Fig. 3 reports how well the amplitude (and intensity) is preserved in general. The coefficient of determination R^2 in Fig. 4 describes how well the values match the linear regression function, while the RMSE in Fig. 5 informs about the noise level. Finally, the standard deviation of the phase difference in Fig. 6 shows how well the phase is preserved. The median of the amplitude difference and the RMSE were normalized by dividing the result by the standard deviation of the original image amplitude.

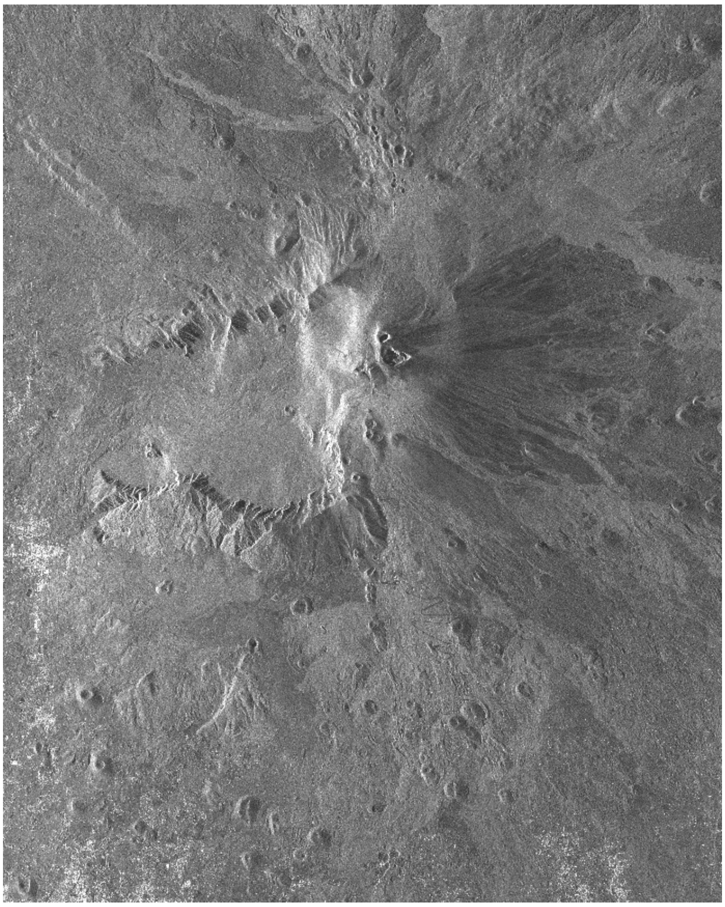

Fig. 1 TerraSAR-X image, Etna, Sicily, Italy

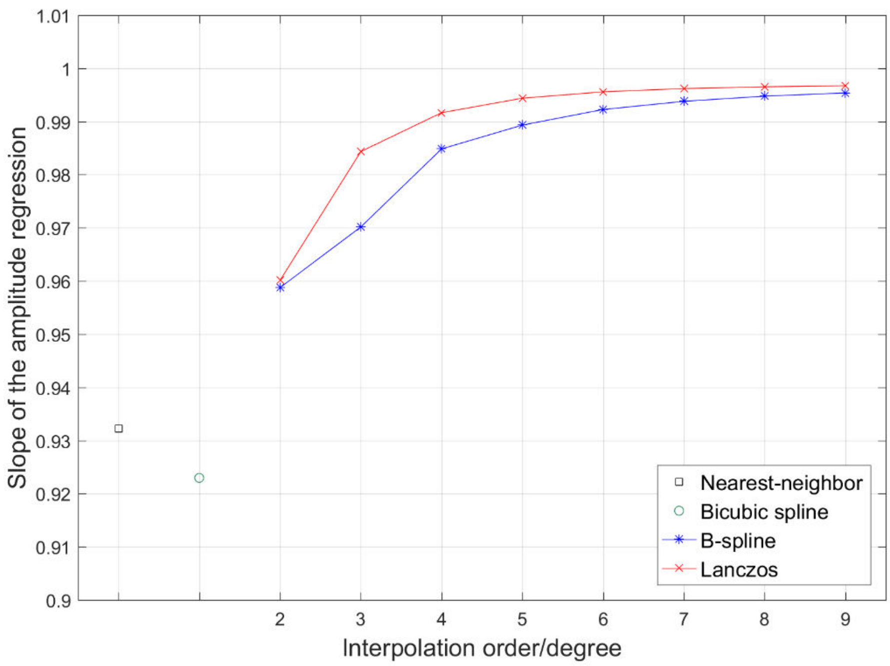

Fig. 2 Slope of the amplitude regression in the first rotation experiment (no oversampling).

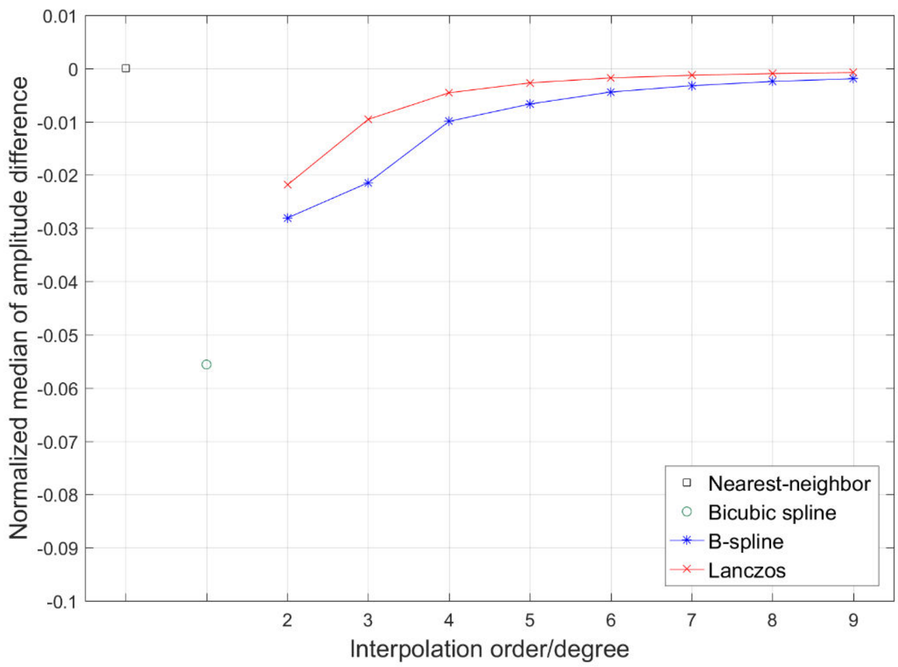

Fig. 3 Normalized median of the amplitude difference in the first rotation experiment (no oversampling).

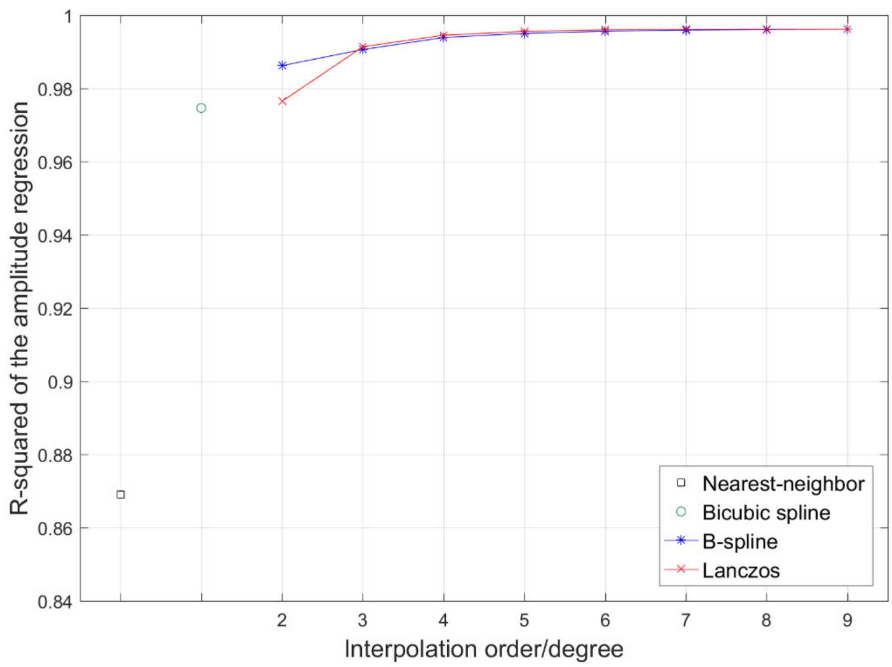

Fig. 4 Coefficient of determination (R^2) of the amplitude regression in the first rotation experiment (no oversampling).

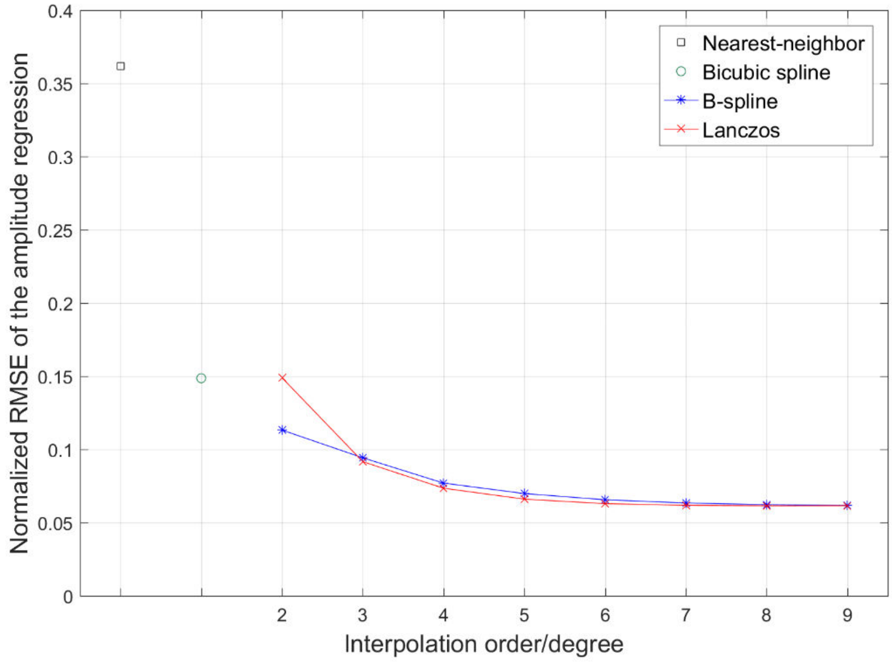

Fig. 5 Normalized root-mean-square error (RMSE) of the amplitude regression in the first rotation experiment (no oversampling).

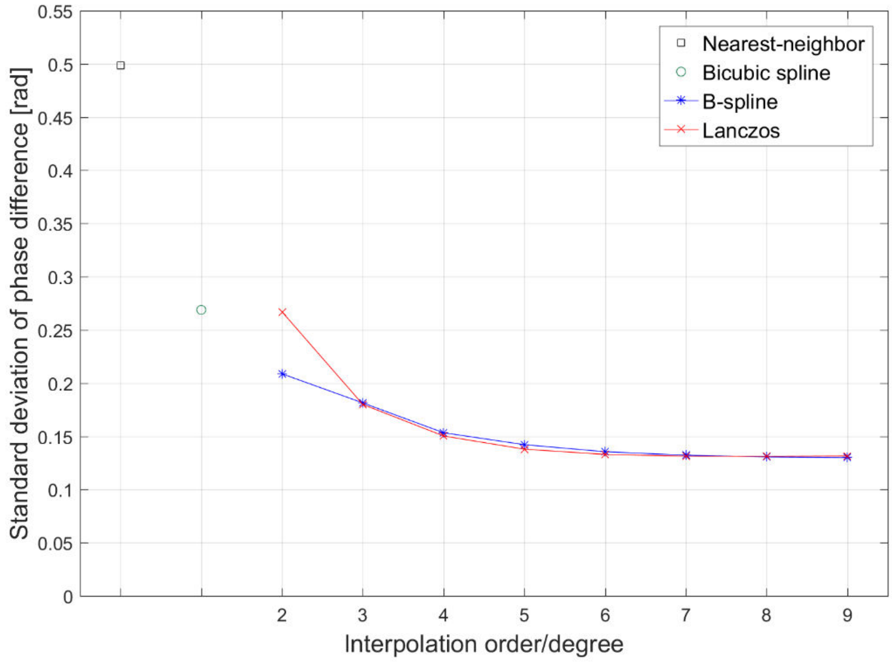

Fig. 6 Standard deviation of the phase difference in the first rotation experiment (no oversampling).

In a second experiment, the same TerraSAR-X SLC was first oversampled by a factor 2 (using a B-spline order 9 interpolation), to rule out any risk of undersampling during the rotations. 45° and -45° rotations were again successively performed using all available interpolation methods and orders. After both rotations, the data were downsampled back to the original sampling (again using a B-spline order 9 interpolation) and compared to the original data. The same statistical values were computed as in the first experiment and are reported in Fig. 7 to Fig. 11.

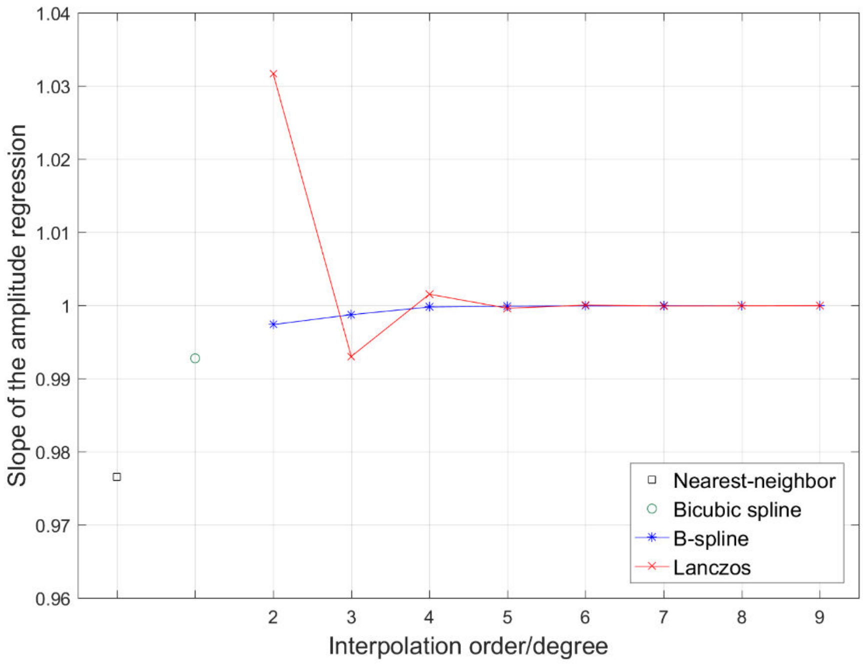

Fig. 7 Slope of the amplitude regression in the second rotation experiment (2x oversampling).

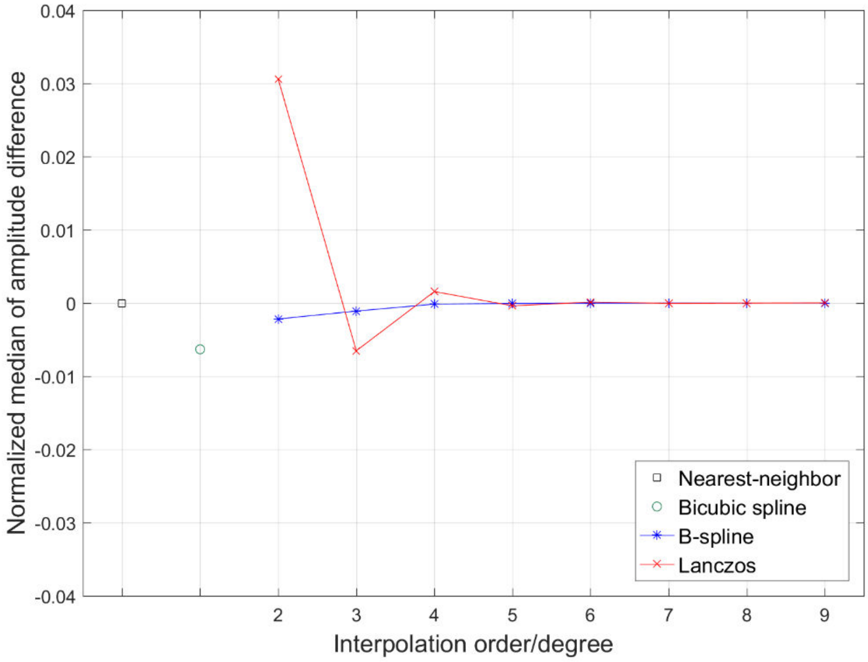

Fig. 8 Normalized median of the amplitude difference in the second rotation experiment (2x oversampling).

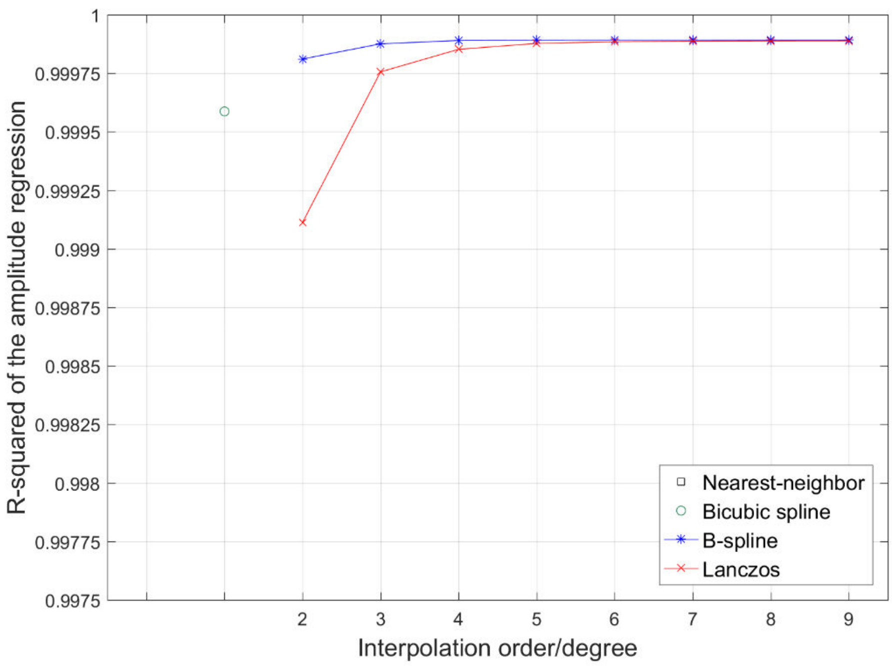

Fig. 9 Coefficient of determination (R^2) of the amplitude regression in the second rotation experiment (2x oversampling).

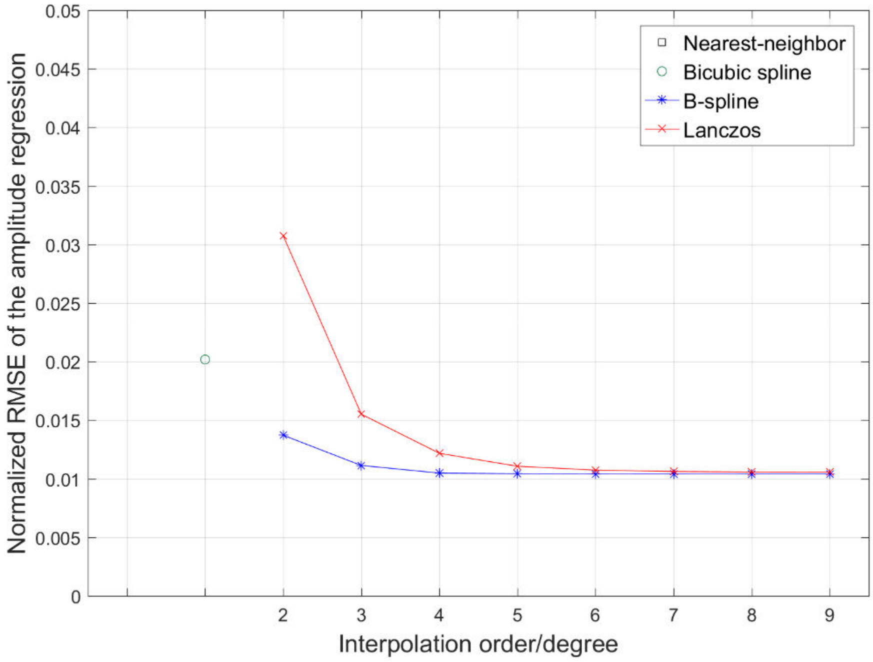

Fig. 10 Normalized root-mean-square error (RMSE) of the amplitude regression in the second rotation experiment (2x oversampling).

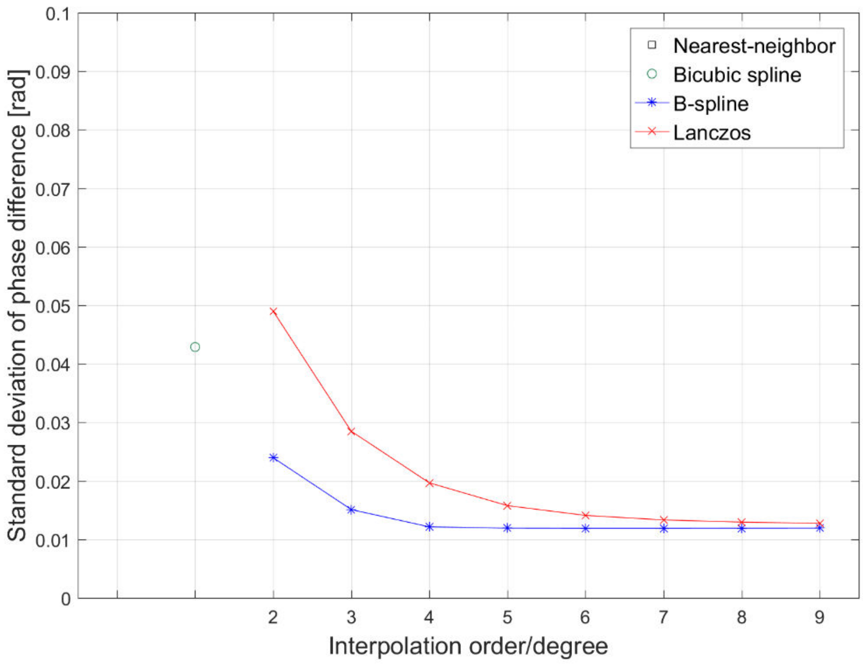

Fig. 11 Standard deviation of the phase difference in the second rotation experiment (2x oversampling).

4.1.2 Intensity SAR data

The rotation experiment was performed again using a MLI image generated from the same TerraSAR-X scene as in Section 4.1.1 (4 looks in each direction). As for the SLC data, the experiment was carried out once by directly rotating the MLI image (Fig. 12 to Fig. 14), and once by rotating the 2-times oversampled image (Fig. 15 to Fig. 17). Note that the intensity was transformed into an amplitude to make it comparable with the results of the previous section. Fig. 18 shows ringing effects caused by inappropriate interpolation (interpolation on the intensity of undersampled data).

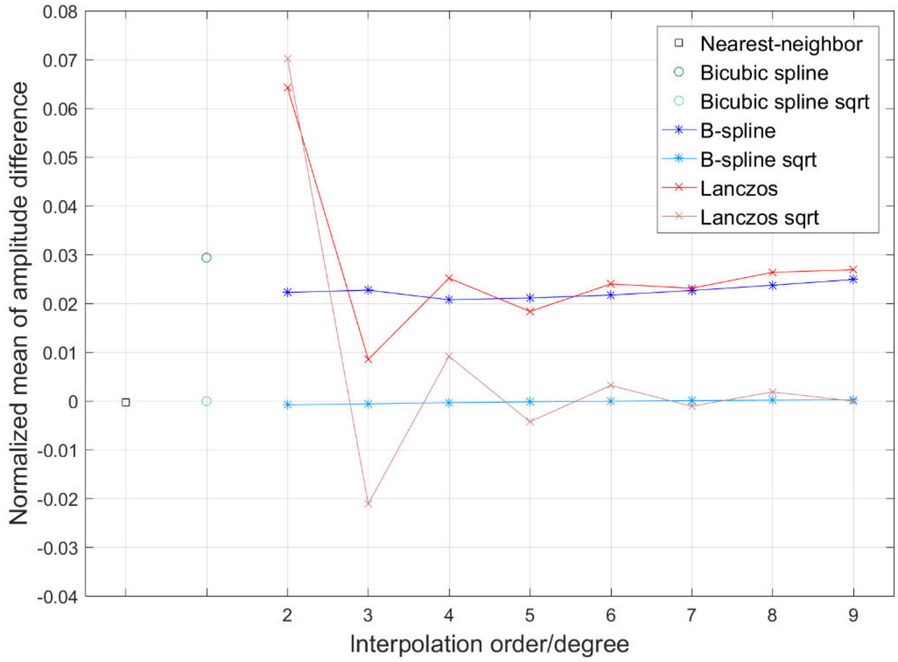

Fig. 12 Normalized mean of the amplitude difference in the first rotation experiment (no oversampling).

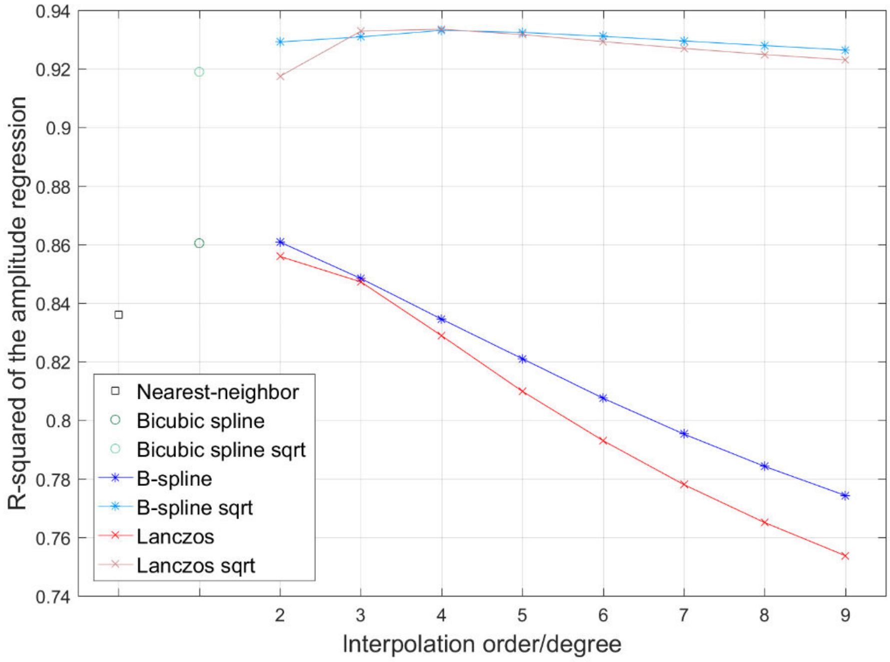

Fig. 13 Coefficient of determination (R^2) of the amplitude regression in the first rotation experiment (no oversampling).

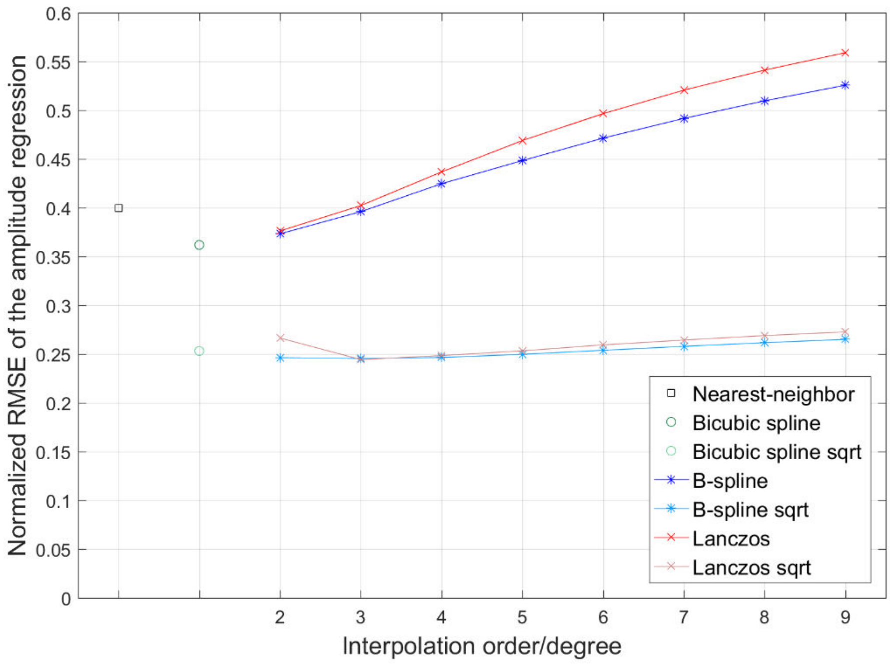

Fig. 14 Normalized root-mean-square error (RMSE) of the amplitude regression in the first rotation experiment (no oversampling).

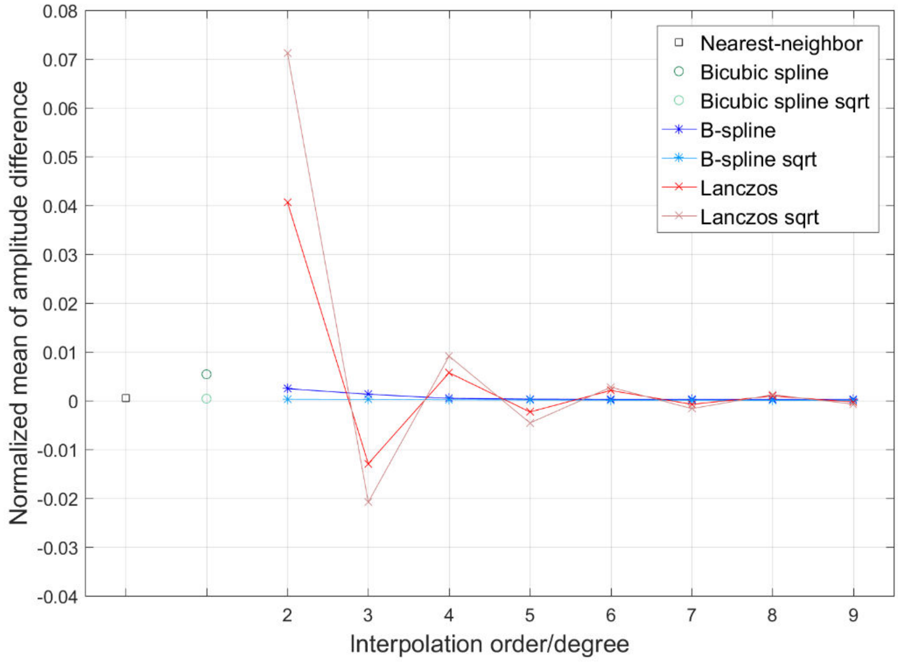

Fig. 15 Normalized mean of the amplitude difference in the second rotation experiment (2x oversampling).

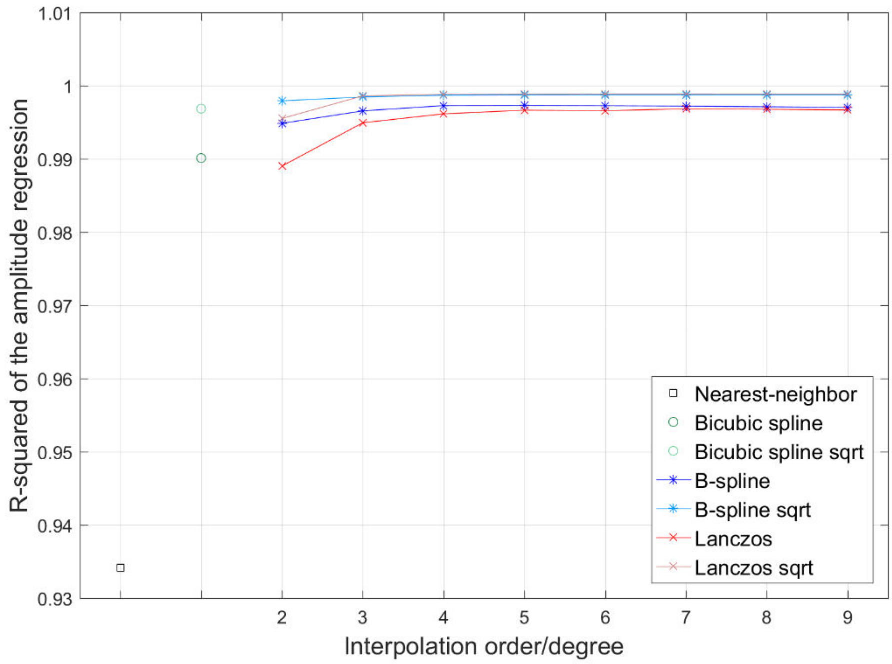

Fig. 16 Coefficient of determination (R^2) of the amplitude regression in the second rotation experiment (2x oversampling).

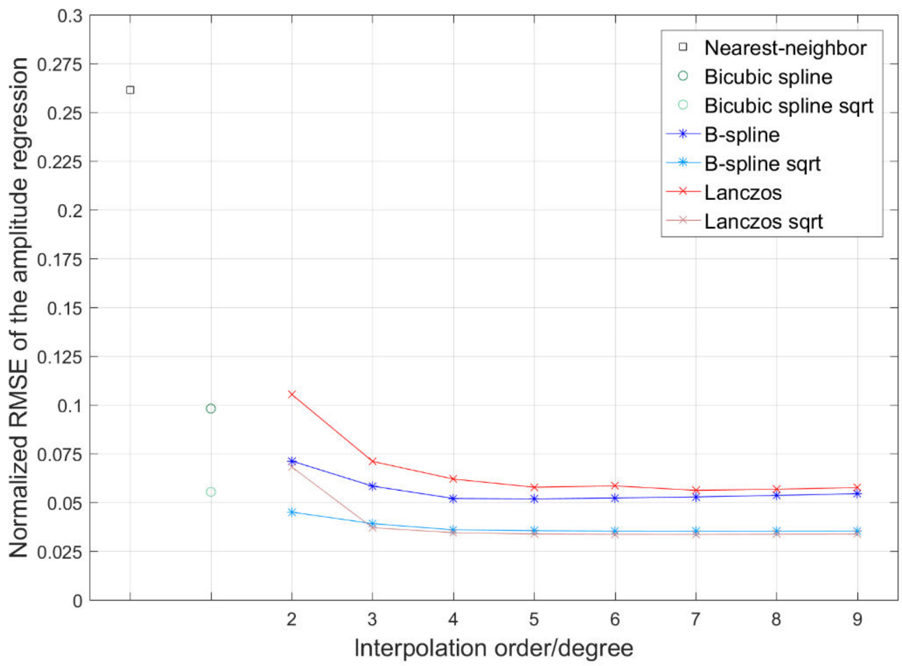

Fig. 17 Normalized root-mean-square error (RMSE) of the amplitude regression in the second rotation experiment (2x oversampling).

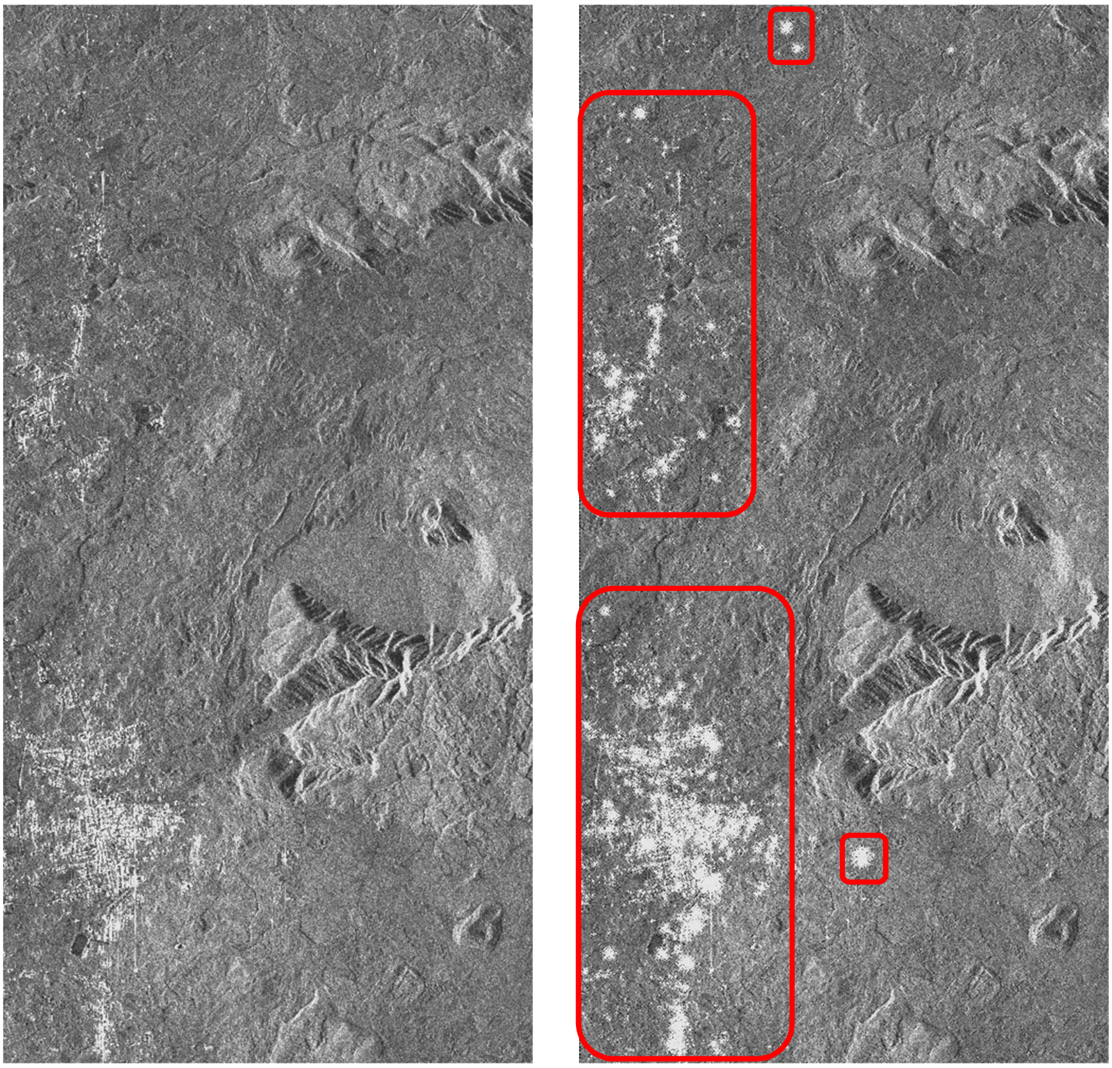

Fig. 18 Left: part of the original MLI image - Right: result of the rotation experiment (B-spline order 9, no oversampling, no square root). Areas with major ringing artifacts are delineated in red.

4.2 DEM and simulated data interpolation

An experiment was performed where a digital surface model (DSM) was transformed from Swiss coordinates into WGS84 equiangular (EQA) coordinates, and back. The pixel spacing in the intermediate product was selected to be approximately the same as in the original data. The transformation was performed using *map_trans* program, i.e. it treated the DSM as a float image (*dem_trans* only has a limited number of interpolation possibilities).

Lanczos interpolation was quickly discarded, as the lack of the partition of unity property causes large artifacts in smooth areas (see mean of the height difference in Fig. 20).

Two issues were then investigated: ringing effects at the edges of vertical structures / sharp edges in the transformed data (Fig. 19), and how well the original data has been preserved after the two conversions (Fig. 20 to Fig. 23).

Fig. 19 DSM showing buildings visualized as shaded relief. It has been converted from Swiss to WGS84 EQA coordinates. Left: using B-spline order 2, right: using B-spline order 9. Ringing effects are visible at the edges of the buildings.

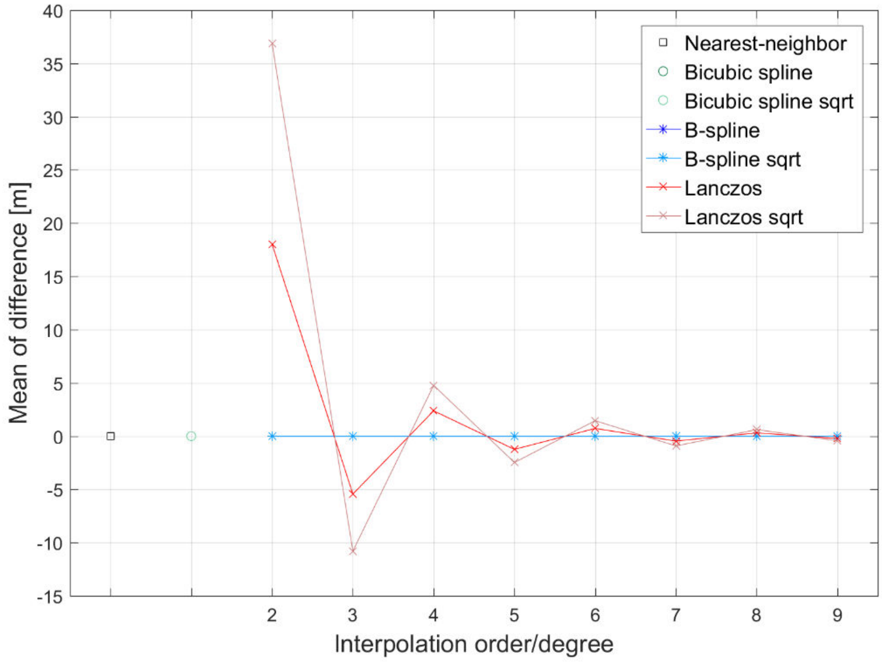

Fig. 20 Mean of the height difference, large scale. The B-spline points are overlaid by B-spline sqrt points at this scale.

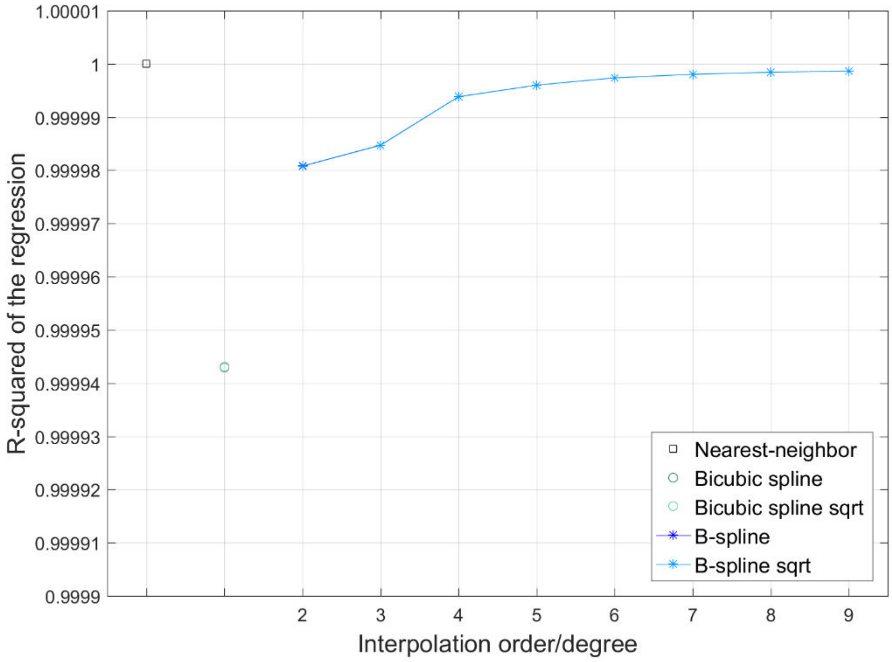

Fig. 21 Coefficient of determination (R^2) of the regression. The B-spline points are overlaid by B-spline sqrt points at this scale.

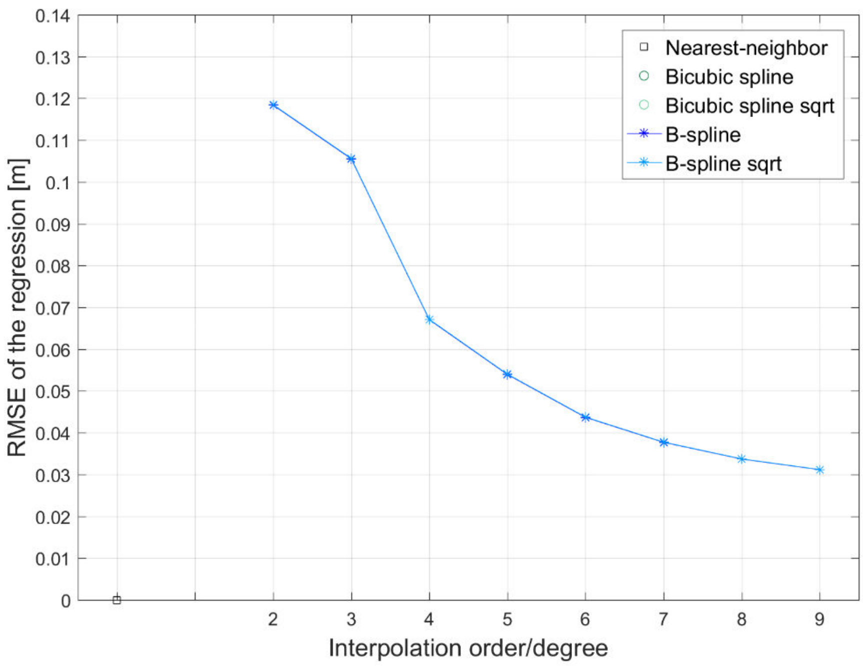

Fig. 22 Root-mean-square error (RMSE) of the regression. The B-spline points are overlaid by B-spline sqrt points at this scale.

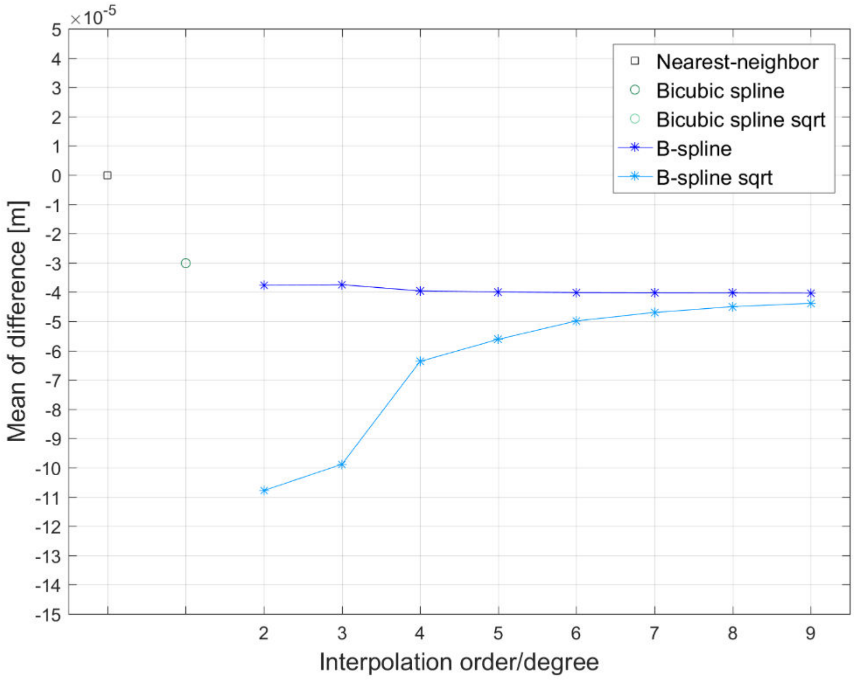

Fig. 23 Mean of the height difference.

5 DISCUSSION

The different cases and the results of the experiments performed in Section 4 are discussed in the following sections. To confirm the results obtained on the rotation of the TerraSAR-X scene, the same experiments were performed on Sentinel-1 TOPS data, with very similar results. Robust conclusions can therefore be drawn from these experiments.

Lanczos interpolation was tested both using "exact" computation of the coefficients and using precomputed coefficients. Only marginal differences could be measured between both methods. Since the use of precomputed coefficients considerably speeds up the calculations, this version is in use in the programs of the Gamma Software.

Processing speed largely depends on the size of the interpolating function (i.e. the interpolation kernel order), and on how many values have to be interpolated compared to the size of the input data. If only few values have to be interpolated, the B-spline interpolation will be particularly slow due to the coefficient calculation. On the other hand, if a large number of values has to be interpolated (larger than the number of input points), Lanczos interpolation will be inefficient due to the large size of the interpolation kernel.

5.1 Complex-valued SAR data

Interpolation of complex-valued data such as SLCs is highly sensitive to the interpolation method. Sub-optimal interpolation leads to loss of information (visible as a decrease of the average intensity), low-pass filtering effects, and phase errors.

Two approaches are possible for this type of interpolation:

- FFT-based oversampling followed by fast interpolation (e.g. using a cubic spline)
- Direct interpolation using high-quality time-domain interpolation method (high order B-spline / Lanczos interpolation)

In the Gamma software, the second approach was preferred.

Lanczos and B-spline interpolation methods both provide very high-quality interpolation for single look complex (SLC) data. In both cases, the higher the order, the better the interpolation for SLC data, at the cost of longer processing time. A 4^{th} order interpolation kernel seems to be a minimum requirement; in the case of Lanczos interpolation, order 2 should never be used, and order 3 is not recommended.

As seen in Fig. 2 and Fig. 3, Lanczos interpolation tends to better preserve the amplitude (and hence the intensity) for data sampled close to the Nyquist criterion, for a given interpolation kernel order. It is not the case for oversampled data, where Lanczos lack of the partition of unity property becomes the main error source.

The recommendation for most cases is to use a mid to high order Lanczos interpolation. For oversampled data, a mid to high order B-spline interpolation gives the best results.

5.2 Intensity SAR data

In both cases of undersampled and non-undersampled data, results show that the interpolation should always be performed on the square root of the intensity (i.e. on the amplitude), indifferent of the interpolation method. The benefit is particularly large on undersampled data, where the interpolation on the square root prevents most of the ringing artifacts (Fig. 12). Notice that use of the square root method is not adequate for data with negative values.

Results also show that the B-spline interpolation does not introduce any significant bias, while the results of the Lanczos interpolation tend to have a bias linked to the interpolation kernel order (Fig. 12 and Fig. 15). The higher the order, the lower the bias. This bias is again probably linked to the partition of unity property not being fulfilled by the Lanczos interpolation kernel.

Note that high-order Lanczos or B-spline interpolation does not significantly improve the results compared to mid-order (4-5) B-spline interpolation, it may even worsen them in case of undersampling by introducing ringing artifacts (Fig. 13 and Fig. 14).

Since low-to-mid order Lanczos interpolation suffers from the bias issue, and both high-order Lanczos and B-spline interpolation risk introducing ringing artifacts, a mid-order (4-5) B-spline interpolation is therefore recommended.

5.3 Interpolation of DEMs or smooth data with abrupt changes

Interpolation of DEMs or smooth data with abrupt changes, such as e.g. backscatter simulations, is a special case. The signal is not bandlimited or periodic, and a sinc-like interpolation method will cause ringing effects at the sides of vertical structures / sharp edges (see Fig. 19). Although the data are generally better preserved using high order interpolation (see Fig. 21 to Fig. 23), the ringing effects are not welcome in the case of DEM interpolation. Smoother, lower “quality” interpolation is less affected, and therefore a cubic spline or B-spline of order 2 to 5 is recommended. Performing the interpolation on the square root of the input data does not bring any benefit, and does not work for data with negative values.

The Lanczos interpolation should not be used at all for such data, because the lack of the partition of unity property causes large artifacts in smooth areas. These are particularly large when using low order Lanczos interpolation (Fig. 20).

6 REFERENCES

[1] P. Thevenaz, T. Blu and M. Unser, "Interpolation revisited [medical images application]," in *IEEE Transactions on Medical Imaging*, vol. 19, no. 7, pp. 739-758, July 2000.

[2] M. Unser, A. Aldroubi, and M. Eden, "B-spline signal processing: Part I theory," *IEEE Trans. Signal Process.*, vol. 41, no. 2, pp. 821–832, Feb. 1993.

[3] M. Unser, A. Aldroubi, and M. Eden, "B-spline signal processing: Part II efficient design and applications," *IEEE Trans. Signal Process.*, vol. 41, no. 2, pp. 834–848, Feb. 1993.

[4] C. Lanczos, "Trigonometric interpolation of empirical and analytical functions," *Studies in Applied Mathematics*, vol. 17, no. 1-4, pp. 123-199, 1938.