11/26/2004

An Implementation of the Fourier Rebinning (FORE) Algorithm into the STIR Framework

GERMAN
CANCER RESEARCH CENTER
IN THE HELMHOLTZ ASSOCIATION

Topics

- The Fourier Rebinning (FORE) Algorithm.
- Example : FORE Rebinning of Line Source Data.

  How FORE Rebinning is done practically.
- Implementation of the FORE Rebinning into the STIR Framework.

  Some Implementation Details and Software Organization.
- Profiling and Run Time Performance.
- Resulting Image Quality and Resolution.

Preface

- I wish to emphasize that the implementation of the FORE rebinning which I am going to present here is based on C++ code provided to me by K. Thielemans. It was written by

    - M. Egger
    - C. Labbé
    - K. Thielemans

The Fourier Rebinning (FORE) Algorithm

- The FORE rebinning algorithm was first proposed by M. Defrise et. al. in 1997 and published in IEEE Transactions on Medical Imaging, Vol 16, NO. 2, p145-158.

- A rebinning algorithm is one that sorts 3-D data sets (one containing oblique sinograms) into an ordinary 2-D data set which contains one sinogram for each transaxial position.

- This 2-D data set can then be reconstructed using any reconstruction algorithm.

- The main intent behind rebinning algorithms is to speed up the reconstruction of 3-D data sets while maintaining a good image quality which is comparable to direct 3-D reconstruction.

The Fourier Rebinning (FORE) Algorithm

- Rebinning algorithms have been in use before but suffered from strong limitations.

- The simplest rebinning algorithm is the single slice rebinning algorithm (SSRB). In SSRB the oblique LORs (line of response) will be assigned *completely* and *exclusively* to the direct sinogram located closest to the mean z-position of the oblique sinogram.

- SSRB is only a good approximation for LORs with small obliqueness.

- In 1994 an improved rebinning algorithm was proposed, the Multi Slice Rebinning algorithm. (Phys Med Biol 39, 1994, Lewitt, Muehllehner, Karp). Here an oblique LOR contributes to all direct sinograms which are crossed.

- Drawback is an increased susceptibility to noise

The Fourier Rebinning (FORE) Algorithm

- The FORE rebinning is based on an *exact* inversion formula which directly relates oblique sinograms to direct ones in frequency space.

$$\wp(\omega, k, \omega_z, \delta) = e^{-ik\arctan(\frac{\delta\omega_z}{\omega})} \wp\left(\omega\sqrt{1+\frac{\delta^2\omega_z^{\,2}}{\omega^2}}, k, \omega_z, 0\right)$$

$$\Updownarrow$$

$$\wp(\omega, k, \omega_z, \delta) = e^{-i\Delta\Phi} \wp(\omega_\chi, k, \omega_z, 0)$$

$$\alpha = \frac{\delta\omega_z}{\omega} = \frac{\tan(\theta)\omega_z}{\omega}$$

$$\Delta\Phi = k \cdot \arctan(\alpha) = k \cdot \arctan(\frac{\delta\omega_z}{\omega})$$

$$\chi = 1 + O(\alpha^2)$$

Taylor development of alpha leads to SSRB in 0-th order and to the FORE approximation in the 1-th order term.

$$P(\omega, k, z, \delta) \approx P(\omega, k, z - \frac{k\delta}{\omega}, 0)$$

Fourier Rebinning Approximation

The Fourier Rebinning (FORE) Algorithm

$$P(\omega,k,z,\delta) \approx P(\omega,k,z+\Delta z) - \frac{\delta^2}{2\omega}\cdot\frac{\partial}{\partial\omega}\cdot\frac{\partial^2}{\partial z^2}P(\omega,k,z+\Delta z,0)$$

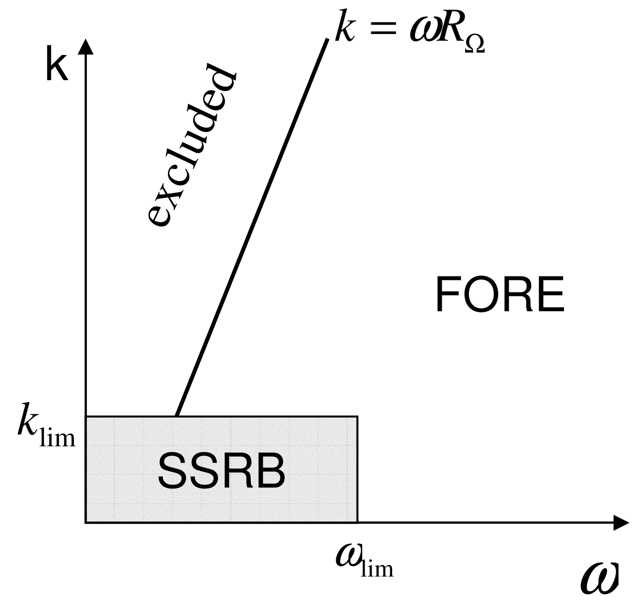
The second order correction will become large if ω gets small. Therefore the first order **FORE approximation can only be applied for large ω.**

➡ "high frequency approximation".

Small ω must be excluded.

The consistency criteria is $|\omega| < \frac{|k|}{R_\Omega}$

R_Ω is the radius of the field of view.



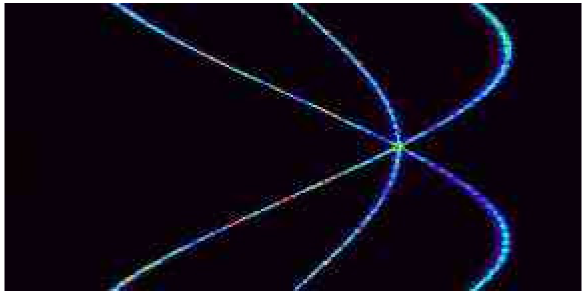
Frequency Distance Relation:

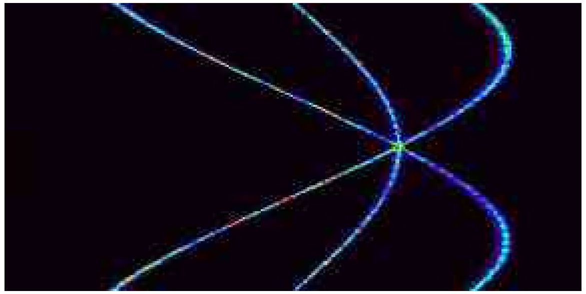
The value of *P* at the frequency (ω,k) receives mainly contributions from sources located at a fixed distance t=-k/ω along the lines of integration.

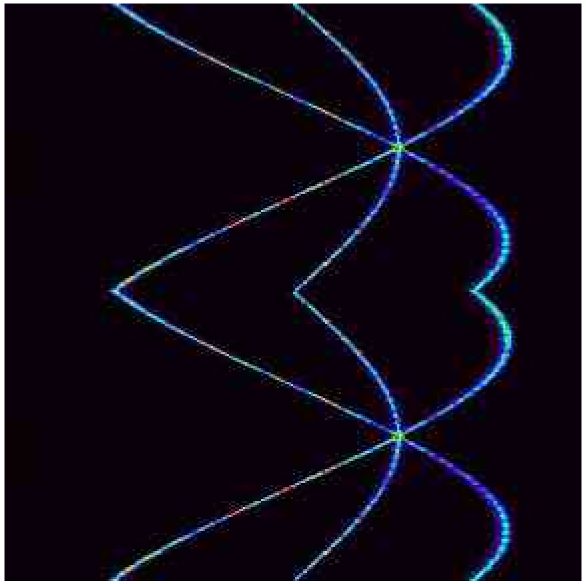
Example: FORE Rebinning of Line Source Data

How to FORE rebin a data set

1. Initialise a stack of 2D rebinned sinograms
2. Consider sequentially each pair of oblique sinograms $(z,\delta);(z,-\delta)$

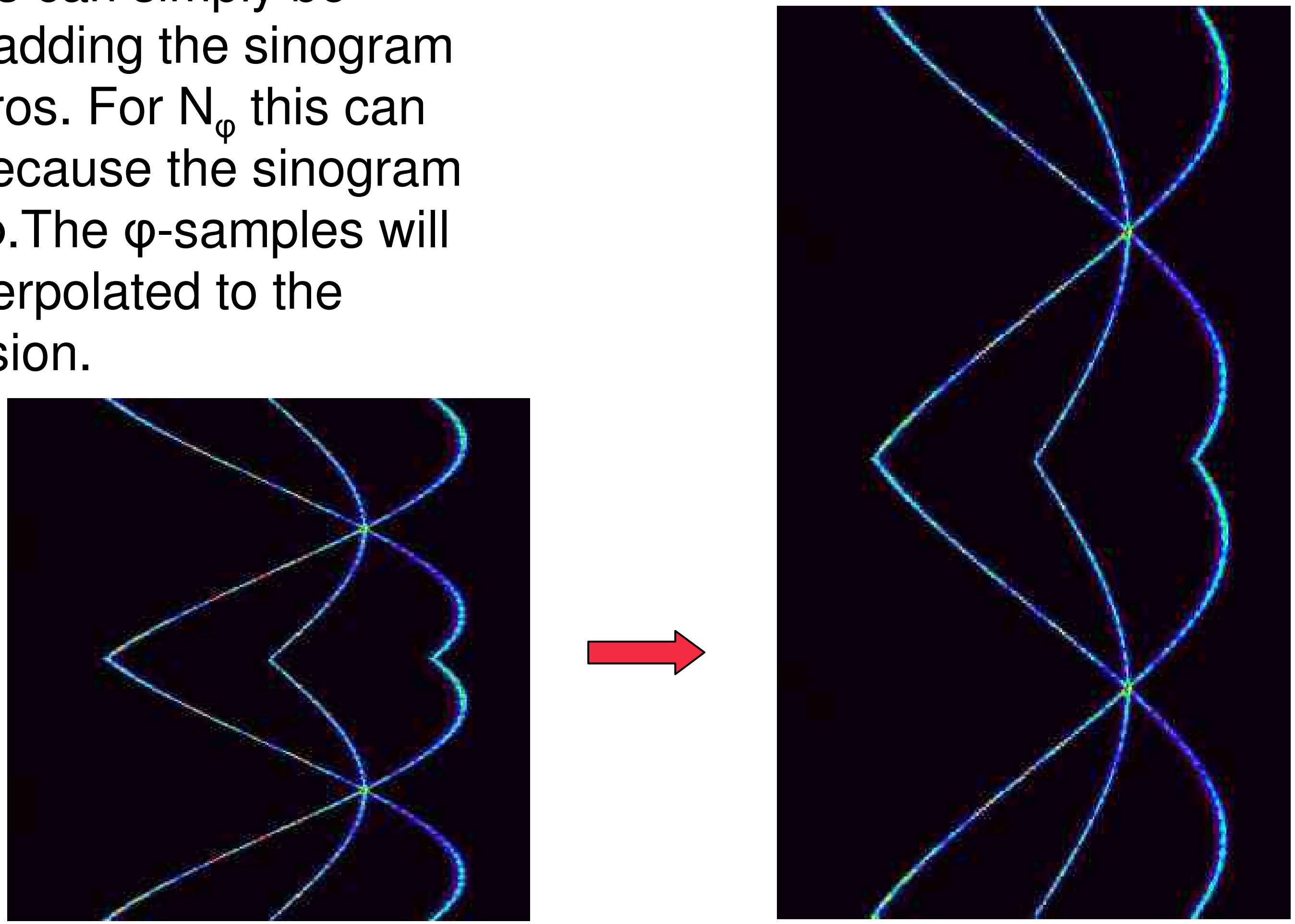






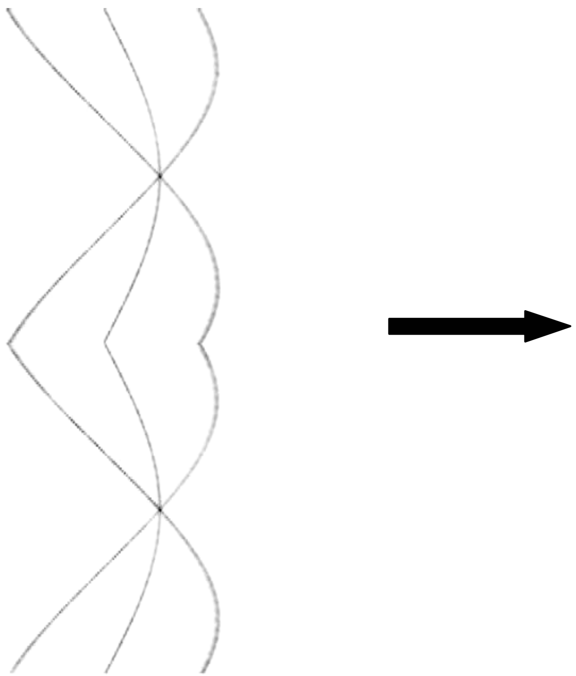
Example: FORE Rebinning of Line Source Data

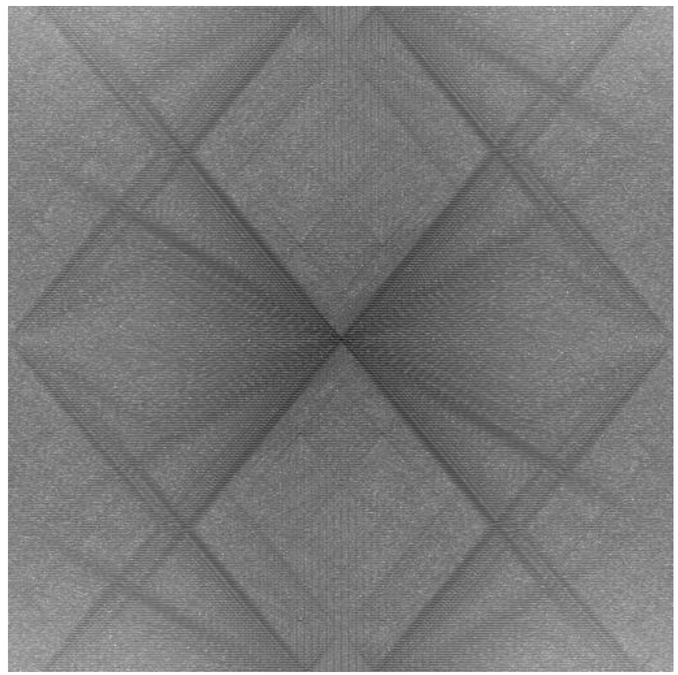
- FFT algorithms are only efficient if N_s and N_φ are equal to powers of two. For N_s this can simply be achieved by padding the sinogram matrix with zeros. For N_φ this can not be done because the sinogram is periodic in φ.The φ-samples will be linearily interpolated to the correct dimension.



Example: FORE Rebinning of Line Source Data

- Calculate the 2-D FFT with respect to s and phi to get $P(\varphi,k,z,\delta)$



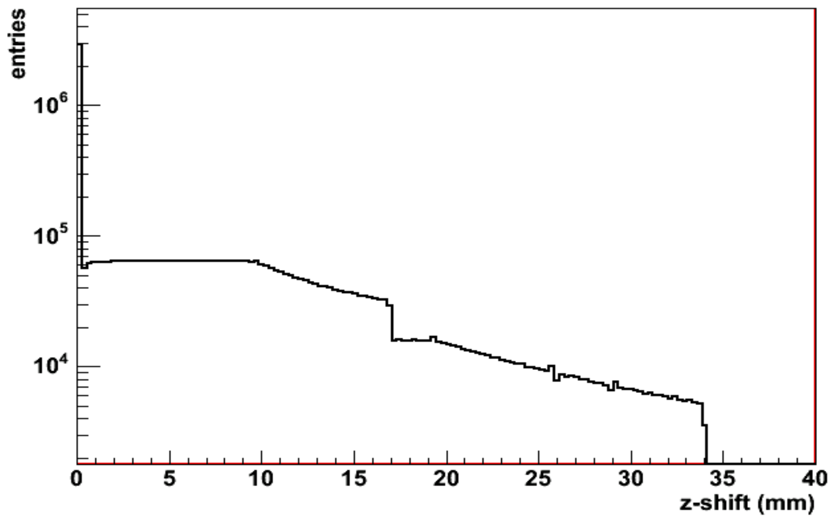


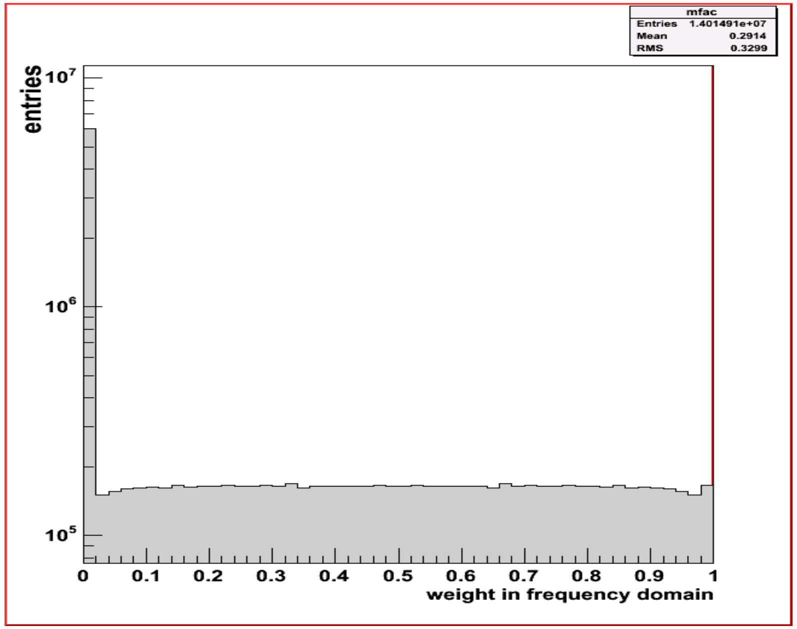
power spectrum

Example: FORE Rebinning of Line Source Data

- For each sample w,k in the FORE or SSRB region the shifted z-position $z' = z - \delta k/\omega$ is calculated.

- The two slices closest to the calculated shifted z-position are identified. The frequency component (ω,k) of the direct sinogram is incremented by the contribution from the oblique sinogram. The contribution to the two neighbouring direct sinograms is multiplied by a weight factor which is determined using linear interpolation.

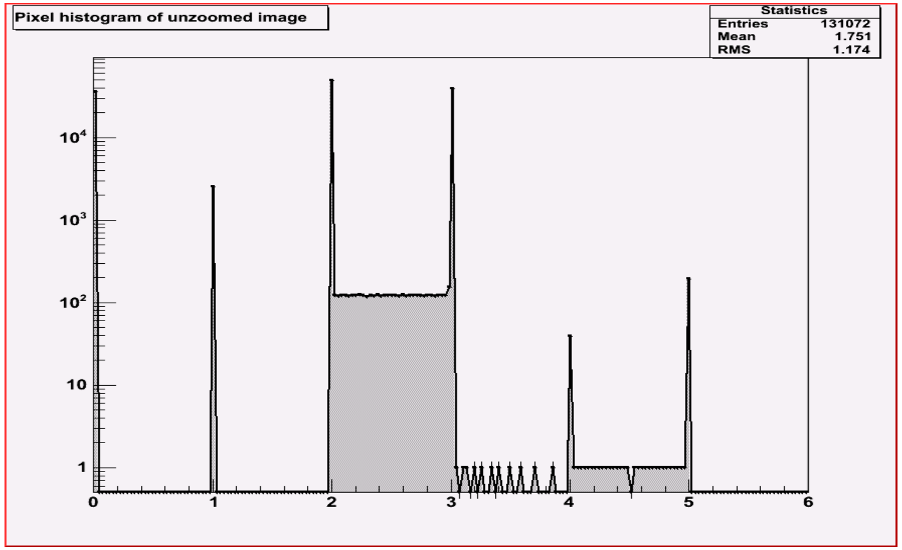




Dr. Oliver Nix
Department for medical physics in radiology
dkfz.

Example: FORE Rebinning of Line Source Data

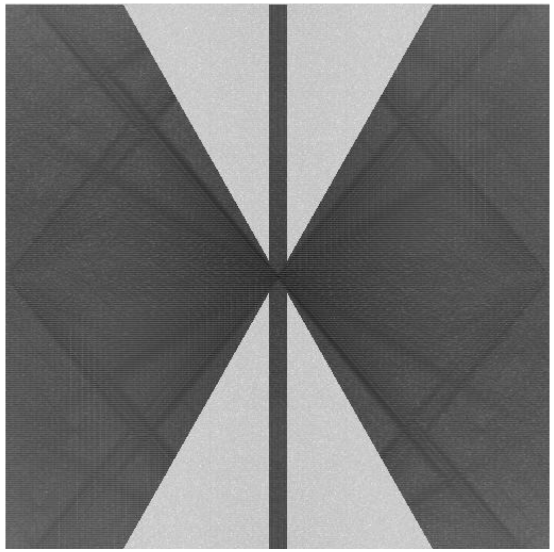
- The weights are accumulated for each frequency component and stored as a weight map. This weight map is needed for normalization of the rebinned data to take into account the variable number of contributions to each frequency component.



- Due to the consistency criteria

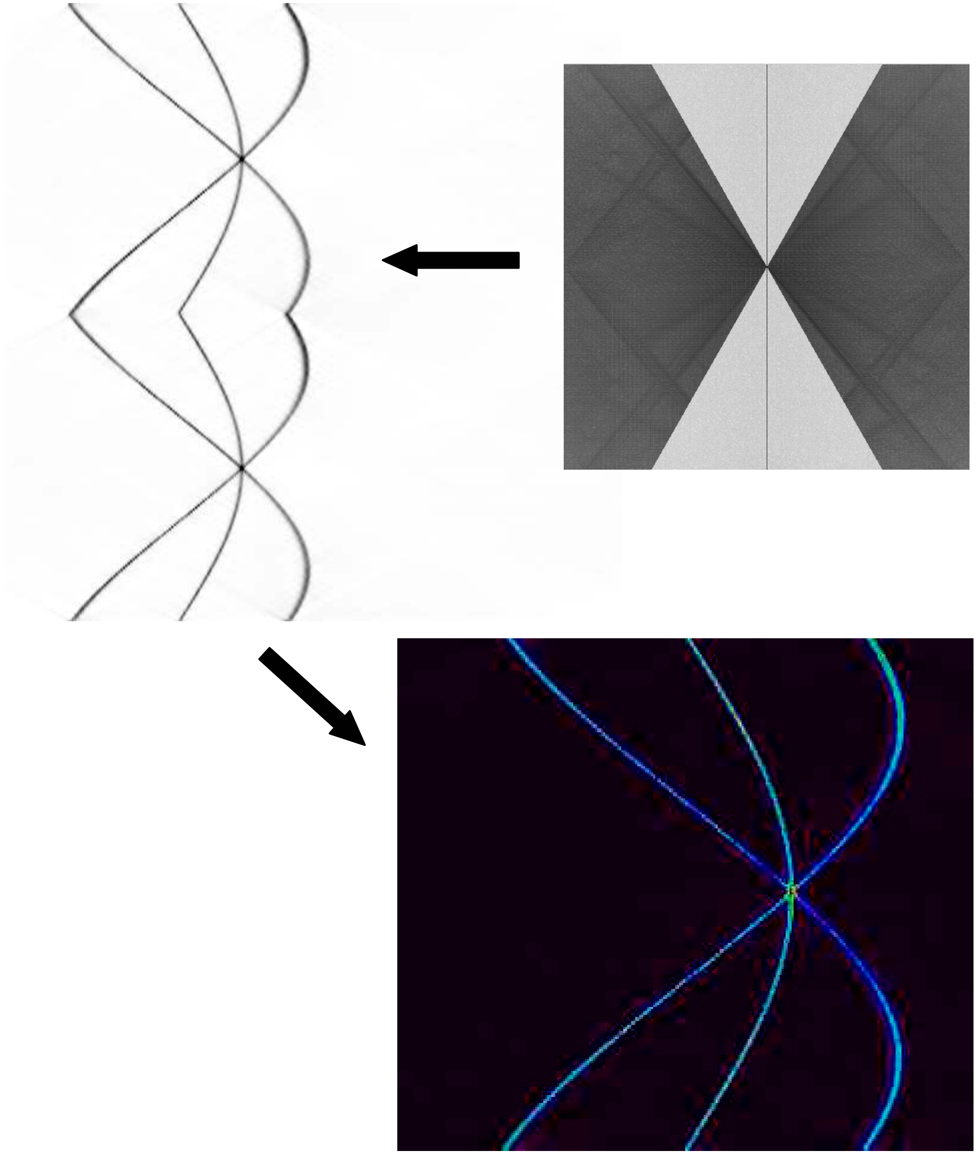
$$|\omega| < \frac{|k|}{R_\Omega}$$

certain frequency components will be forced to zero.



Example: FORE Rebinning of Line Source Data

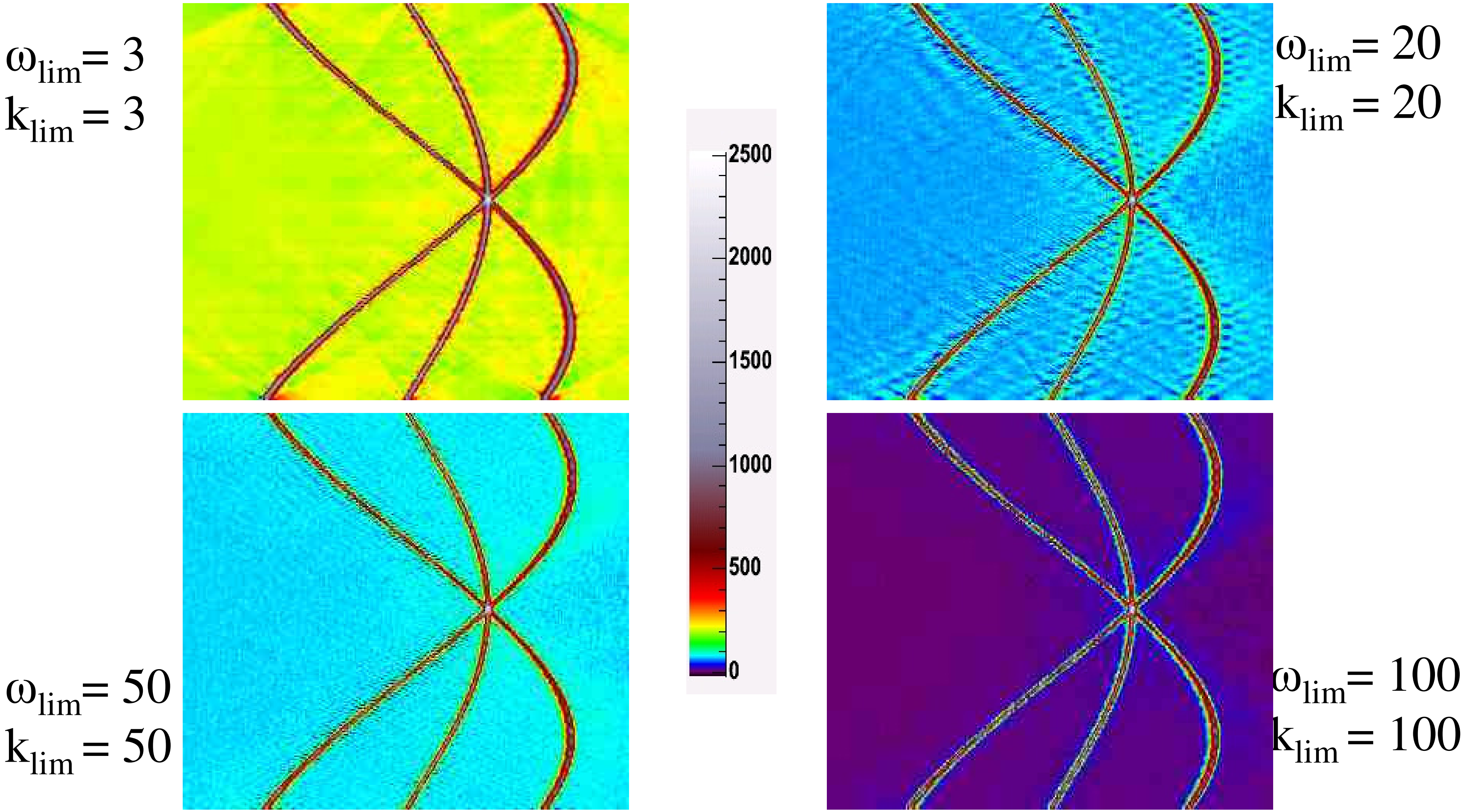
- The inverse 2-D FFT of the rebinned sinograms is calculated.



- The original dimensions (number of tangential positions and views) are restored.
- This data can then be reconstructed using any 2-D reconstruction algorithm.

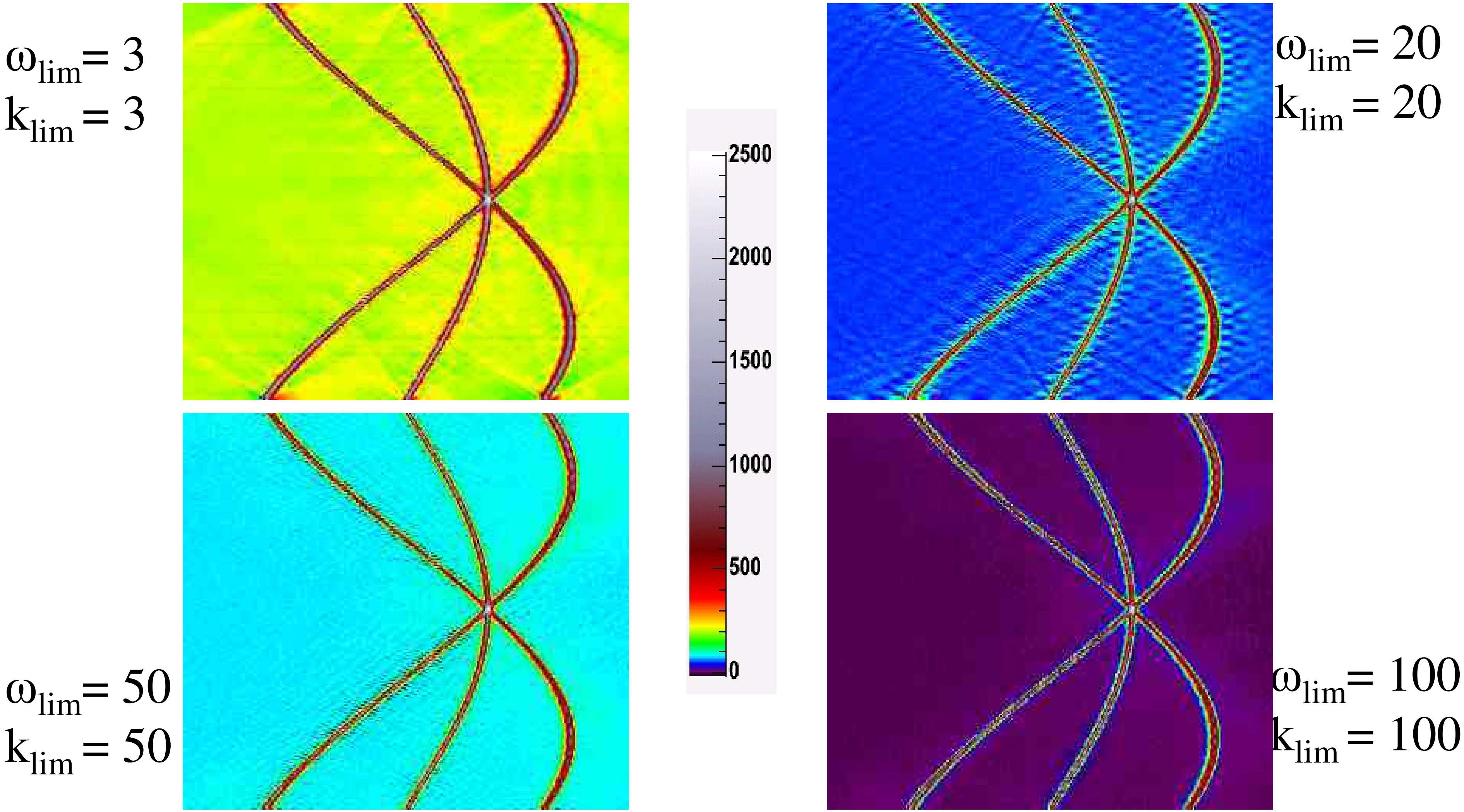
Example: FORE Rebinning of Line Source Data

- Effects of different parameter sets for ω_{lim} and k_{lim} on the rebinned sinograms. δ=30



Example: FORE Rebinning of Line Source Data

- Effects of different parameter sets for ω_{lim} and k_{lim} on the rebinned sinograms. δ=1.



Implementation of the FORE rebinning into the STIR framework

- STIR aims at providing a multi-platform object-oriented library for all data manipulations in tomographic image reconstruction.
- Presently it mainly consists of classes, functions and utilities for 3-D PET image reconstruction.
- STIR relies heavily on the concept of object oriented programming using the C++ programming language.
- It is logically organized in two parts.
- Part one is a library providing building blocks for image / projection data manipulation and reconstruction.
- The second part provides utilities for image reconstruction and necessary file format conversions.

→
- The implementation of the FORE rebinning follows this paradigms.
- The rebinning code itself is encapsulated inside a class hierachy.
- A small example application is provided to rebin data sets.

Implementation of the FORE rebinning into the STIR framework

- The base class for all rebinning algorithms is the **ProjDataRebinning** class.
- It implements some basic tasks common to all rebinning algorithms, such as initialization etc.
- All classes dealing with projection data rebinning need to inherit from this rebinning base class. They must implement the member function rebin which is declared as a pure virtual function in the ProjDataRebinning class.
- In this context the Fourier Rebinning is encapsulated inside a **class FourierRebinning** which is derived from ProjDataRebinning and which implements the method **FourierRebinning::rebin().**
- In order to stay flexible the rebinning needs to read parameter from a parameter file. Therefore this class also needs to inherit from Registered(Parsing)Object.

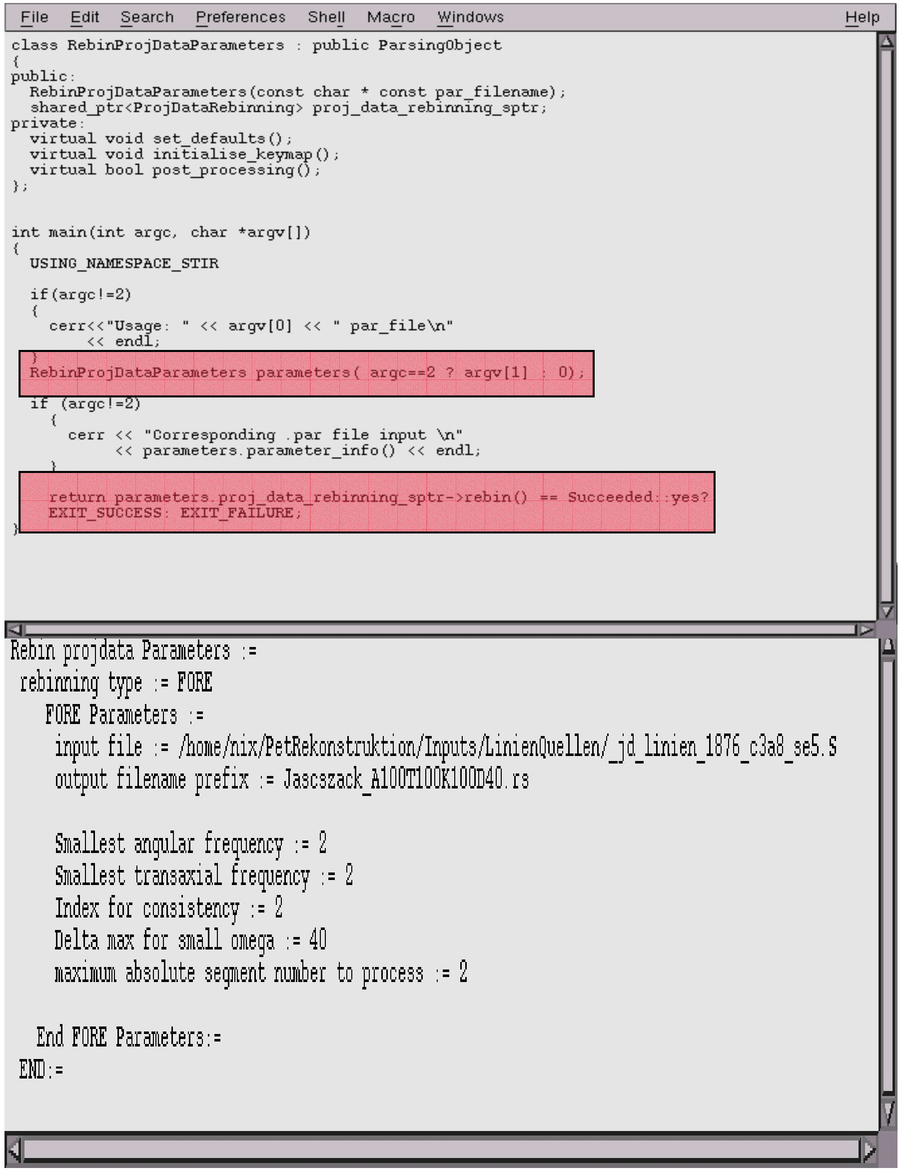
Implementation of the FORE rebinning into the STIR framework

- The fourier rebinning can be run by adding the following lines of code to your reconstruction application or use it in a stand alone application.

- The following parameters need to be passed to the algorithm.
  - Input / Output file names.
  - Definition of the region in frequency space where FORE will be applied.

    $$|\omega| < \frac{|k|}{R_\Omega}$$

  - Delta defines the maximal obliqueness up to which SSRB will be applied for small values of (ω,k)
  - Index for consistency. An additional consistency criteria



Performance

- A set of Jaszczack phantom data acquired with a CTI/Siemens EXACT HR+ PET-Scanner in 3D-mode is used to determine the reconstruction performance and the image quality.
- This data set is arc-, scatter-, and attenuation corrected using CTI/Siemens bkproj_3D (V1.21) application.
- The data is then reconstructed
  - using the bkproj_3D application using the R=1 option (PROMIS)
  - performing a FORE rebinning using the bkproj_3D application and reconstructing the rebinned data with the STIR F2BP, OSMAPOSL and bkproj_3D executable
  - using the OSMAPOSL executable to perform a direct 3D OSEM reconstruction
  - using the OSMAPOSL executable to perform a 2D OSEM reconstruction of the before STIR / bkproj_3D FORE rebinned data sets.

Performance

- The reconstructions using CTI/Siemens software were performed on a SUN Ultra Sparc 2 machine with 256 Mbyte memory and 350 MHz CPU.
- All STIR based reconstructions were done on a SuSe Linux PC with 933 MHz CPU speed and 512 MByte RAM.
- gcc version 3.3.1 was used with the default STIR optimization flags

- Since the bkproj_3D software is not available on Linux and STIR was not installed on the SUN platform (due to the fact that this is a dedicated machine not intended for development purposes) the performance measurements between STIR and CTI based reconstructions cannot be compared directly.

Performance

Rebinning Step	CPU time [s]	rel. CPU time [%]
Overall rebinning	61.9	100
Initialization of data structures. Determination of scanner properties.	2.6*	4.2
Expand the merged sinograms by zero padding and linear interpolation to suitable dimensions for the FFT algorithm.	8.2	13.2
Loop over all transaxial positions and FFT each slice	24.1	38.9
Perform the FORE rebinning	5.5	9.0
Normalize the rebinned sinograms	1.0	1.6
Inverse FFT the rebinned sinograms	11.2	18.3
Restore the original sinogram dimensions.	0.95	1.6

* including restoration of original sinogram dimensions

Performance

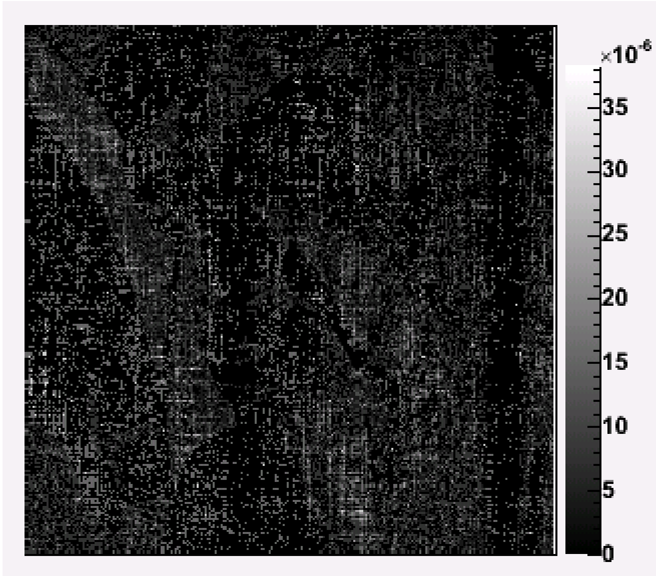
- Approximately 50% of the time needed to rebin a data set is spent in Fourier Transformation of the projection data.

- ➔ FORE performance optimization starts in the FFT part of the code !

- Three FFT implementations were tested for 1 and 2 dimensional FFT
  1. The numerical recipes code (rlft3, fourn)
  2. The FFTW library implementation (see www.fftw.org)
  3. A FFT implementation by K. Thielemans (part of STIR since version 1.3).

Performance

Implementation	Data Type	CPU time
STIR FFT	1D, Array of complex random numbers	0.13
	2D, real data, Lena Image	7.0
	2D, real data, Lena Image, using complex FFT	11.2
NR FFT	1D, Array of complex random numbers	0.16
	2D, real data, Lena Image	4.9
FFTW FFT	1D, Array of complex random numbers	0.04
	2D, real data, Lena Image	7.8



Performance

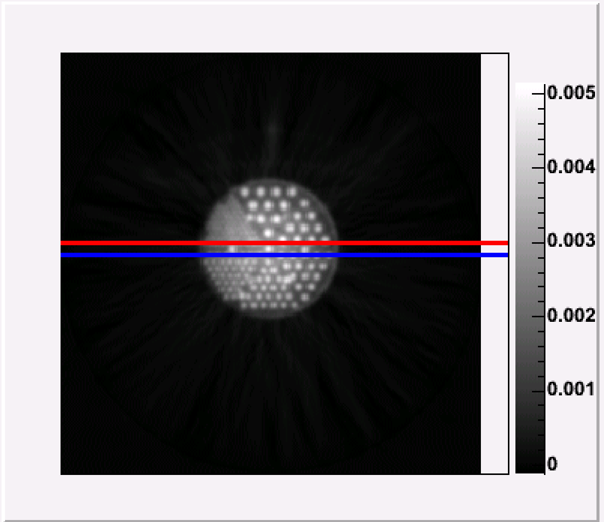
Algorithm	CPU Time [s]
CTI/Siemens bkproj_3D (PROMIS)	3199
CTI/Siemens bkproj_3D (FORE)	1090,3
OSMAPOSL 3D OSEM	1781.3 (9 subsets, 20 iterations) 1190.9 (18 subsets, 20 iterations)
FBP2D – 3-D data set	60.0 (image size 256) 233.6 (image size 512)
STIR – FORE rebinning	61.9
FBP2D – FORE rebinned data set	106.3 (image size 256) 410.1 (image size 512)
OSMAPOSL 2D OSEM	1061 (8 subsets, 20 iterations) 691.4 (16 subsets, 20 iterations) 499.2 (32 subsets, 20 iterations)

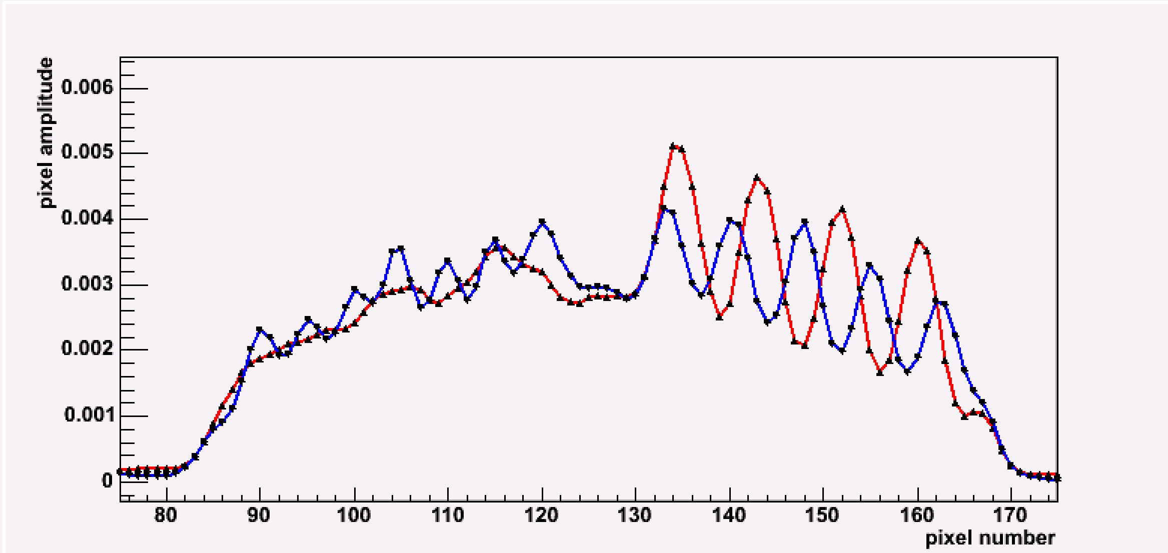
Resolution and Image Quality

- A set of Jaszczack phantom data acquired with a CTI/Siemens EXACT HR+ PET-Scanner in 3D-mode is used to determine the reconstructed image quality and resolution.
- This data set is arc-, scatter-, and attenuation corrected using CTI/Siemens bkproj_3D application.
- The data is then reconstructed
  - using the bkproj_3D application using the R=1 option (PROMIS)
  - performing a FORE rebinning using the bkproj_3D application and reconstructing the rebinned data with the STIR OSMAPOSL executable
  - using the OSMAPOSL executable to perform a direct 3D OSEM reconstruction
  - using the OSMAPOSL executable to perform a 2D OSEM reconstruction of the STIR FORE rebinned data set.
- For comparisons slice 45 is used. The image dimensions is always 256*256. In the following a profile of line 144 and 133 is shown.

Resolution and Image Quality

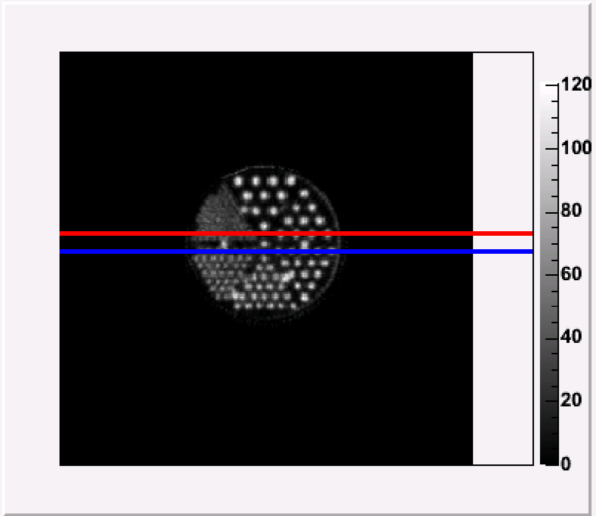
- Reconstructed with the CTI/Siemens Bkproj_3D application using the R=6 option (FORE rebinning) in high resolution mode (-H flag set).

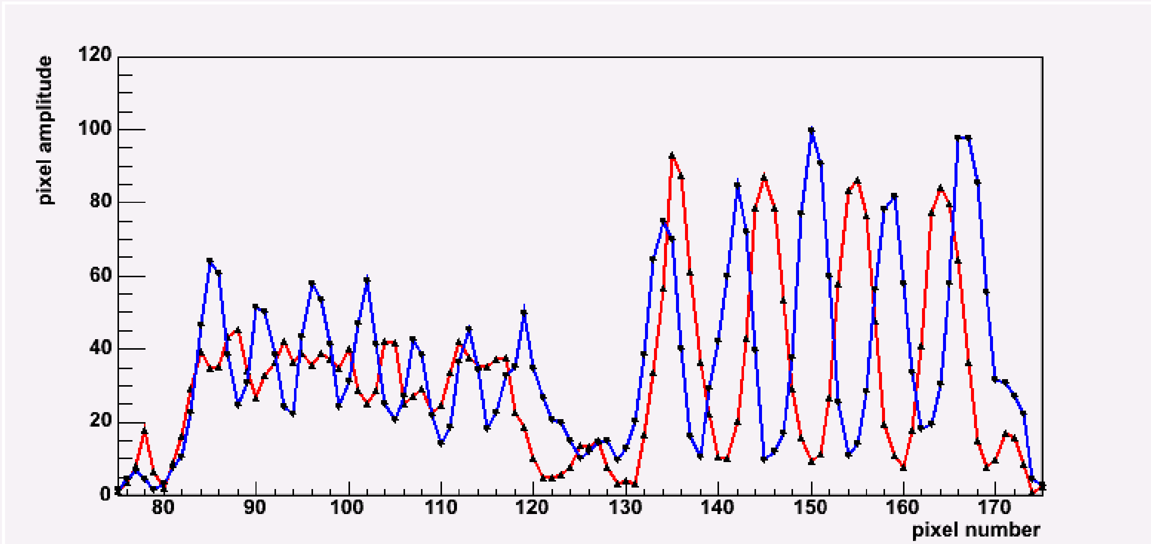




Resolution and Image Quality

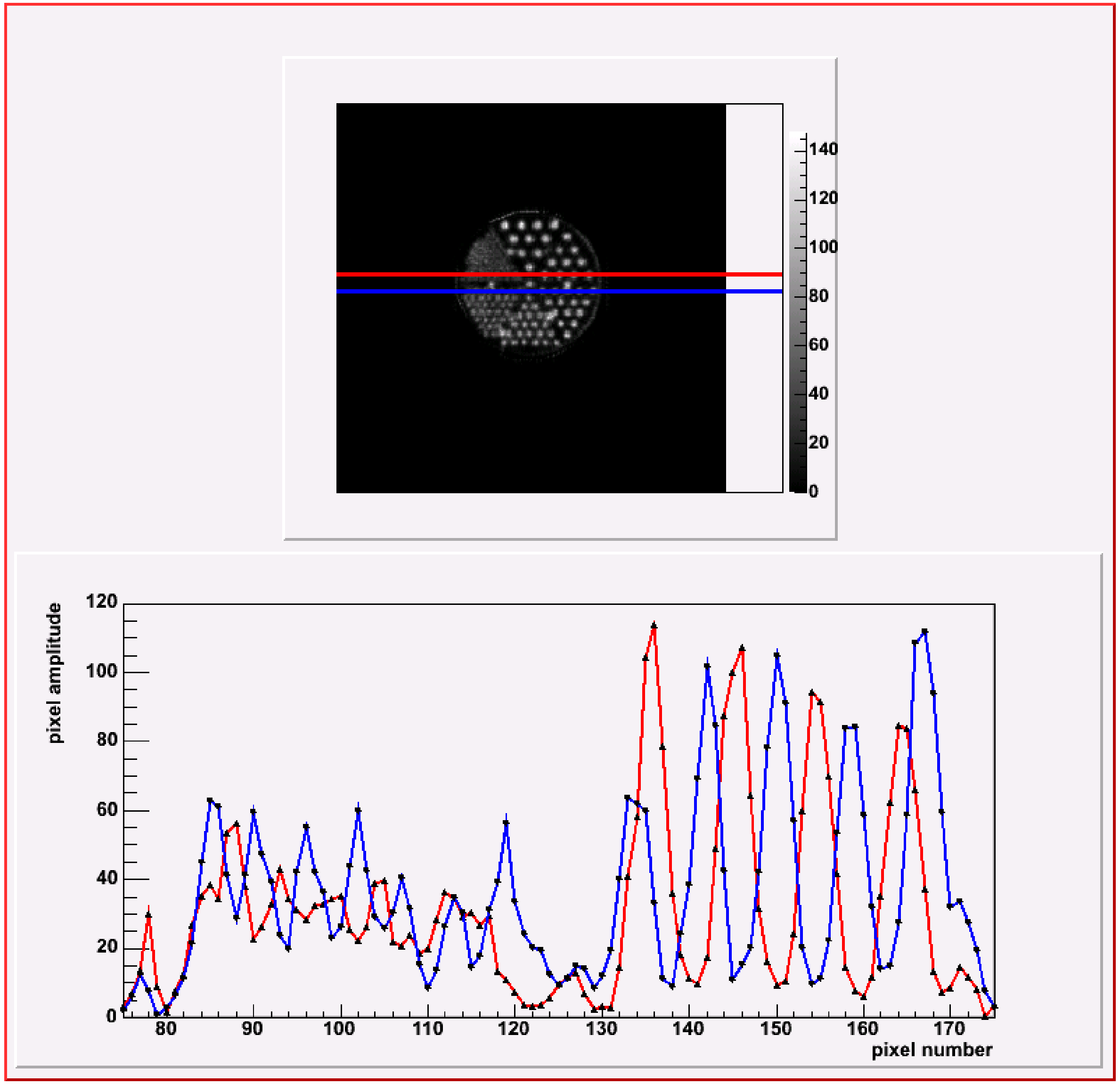
- Reconstructed with the STIR OSMAPOSL executable using 20 iterations and 9 subsets. The sensitivity image was generated by the sensitivity executable and an initial image produced by a simple backprojection.





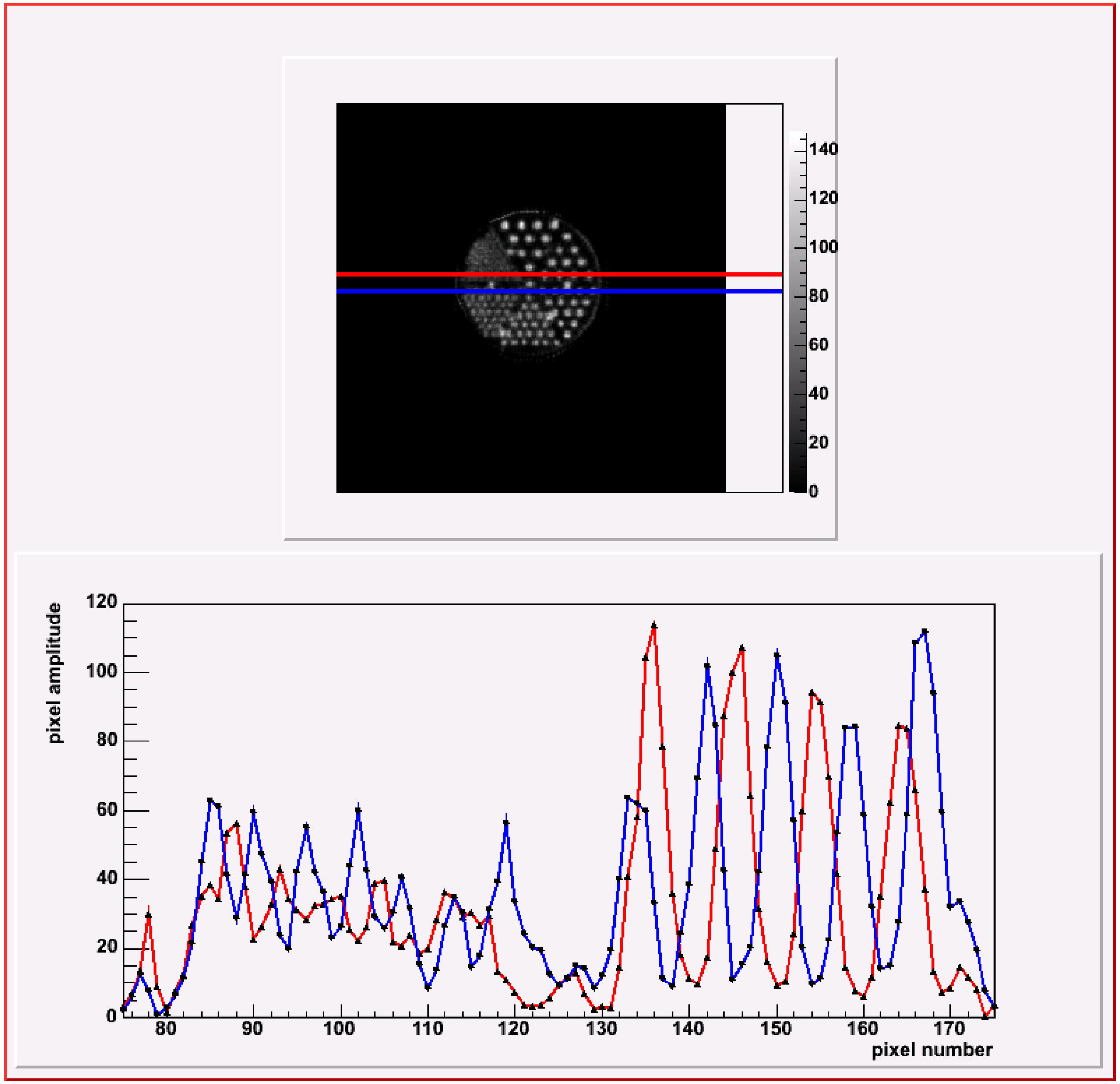
Resolution and Image Quality

- Reconstructed with the STIR OSMAPOSL executable using 20 iterations and 18 subsets. The sensitivity image was generated by the sensitivity executable and an initial image produced by a simple backprojection.



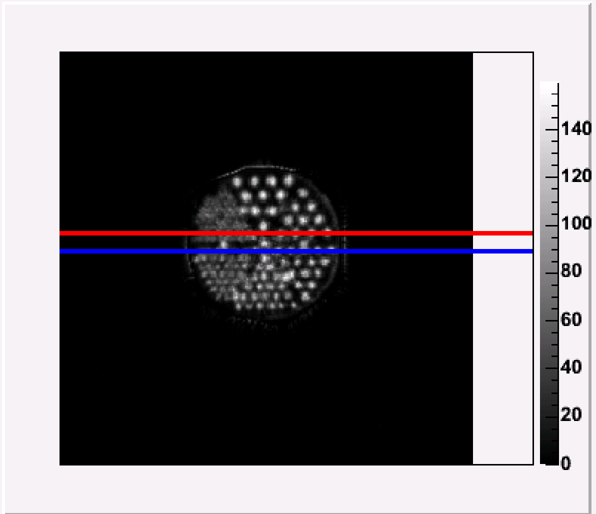
Resolution and Image Quality

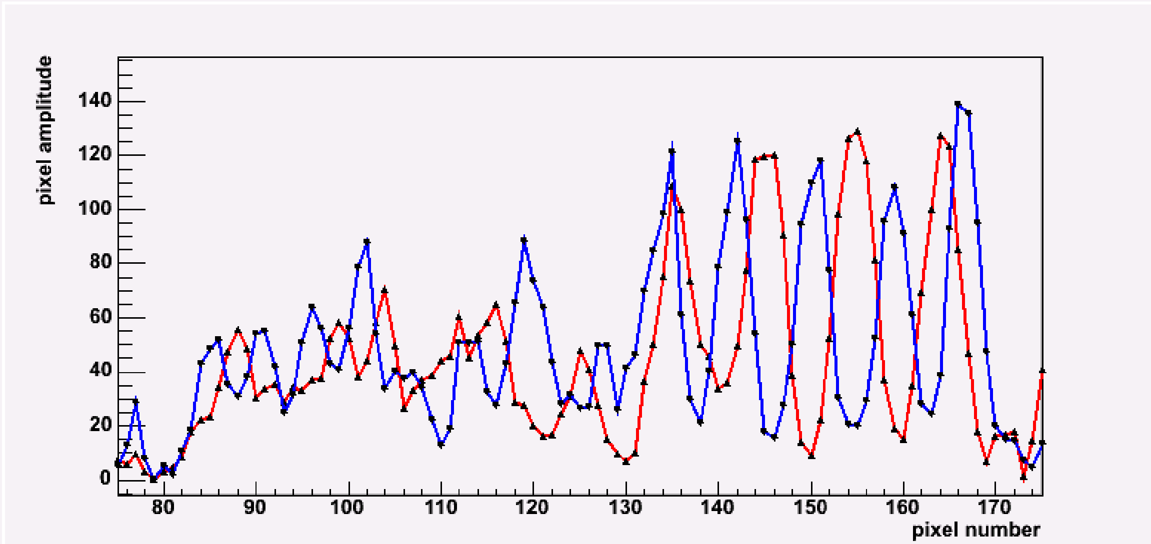
- The 3D data set was rebinned using the STIR FORE implementation with a „standard set“ of parameters
- Smallest angular frequency = 20
- Smallest transaxial frequency = 20
- Consistency Index = 20
- Delta max for rebinning = 1
- OSMAPOSL as before, number of iterations = 20, number of subsets 8.



Resolution and Image Quality

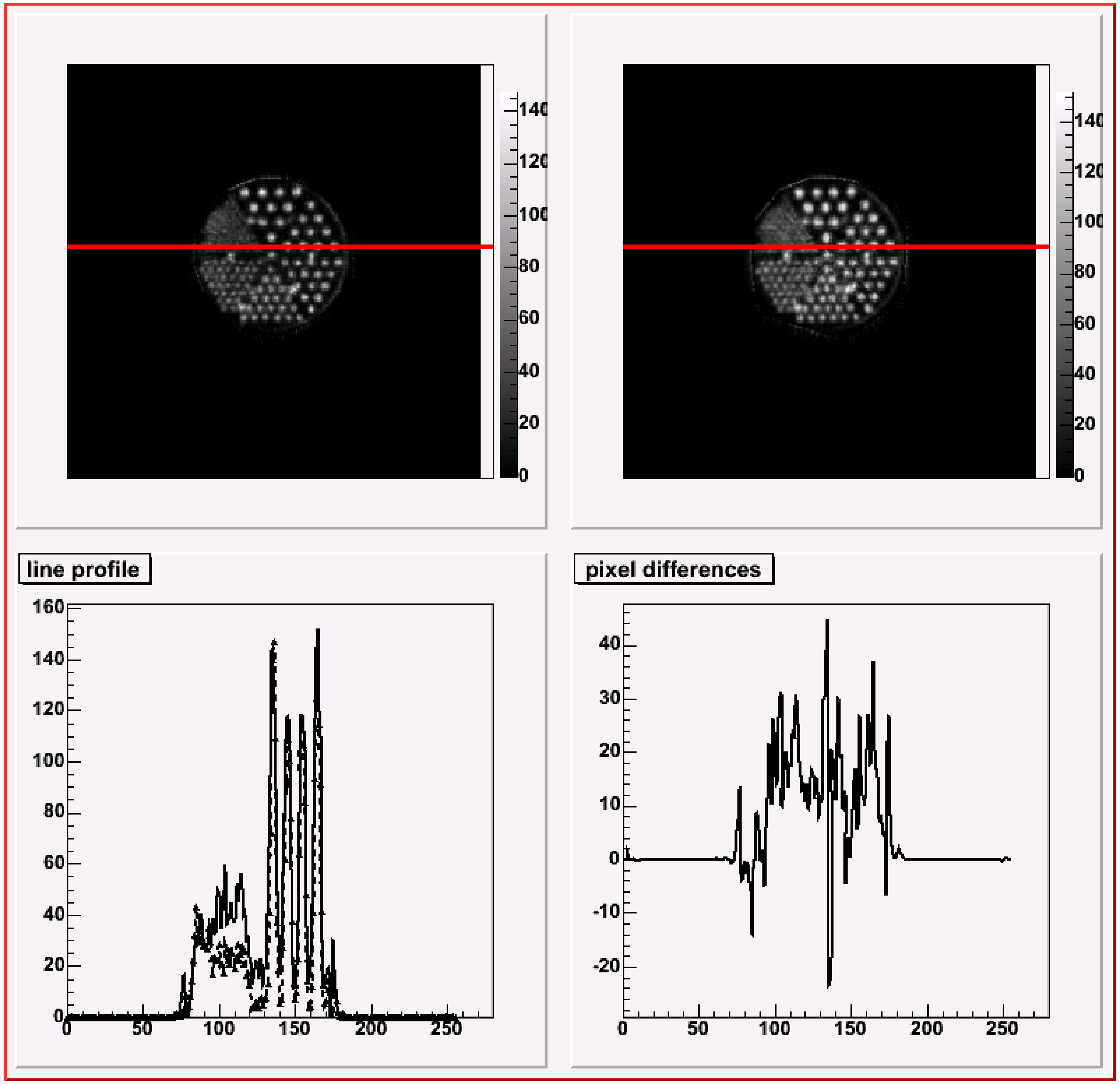
- The 3D data set was rebinned using the STIR FORE implementation with a „standard set“ of parameters
- Smallest angular frequency = 20
- Smallest transaxial frequency = 20
- Consistency Index = 20
- Delta max for rebinning = 1
- OSMAPOSL as before, number of iterations = 20, number of subsets 16.





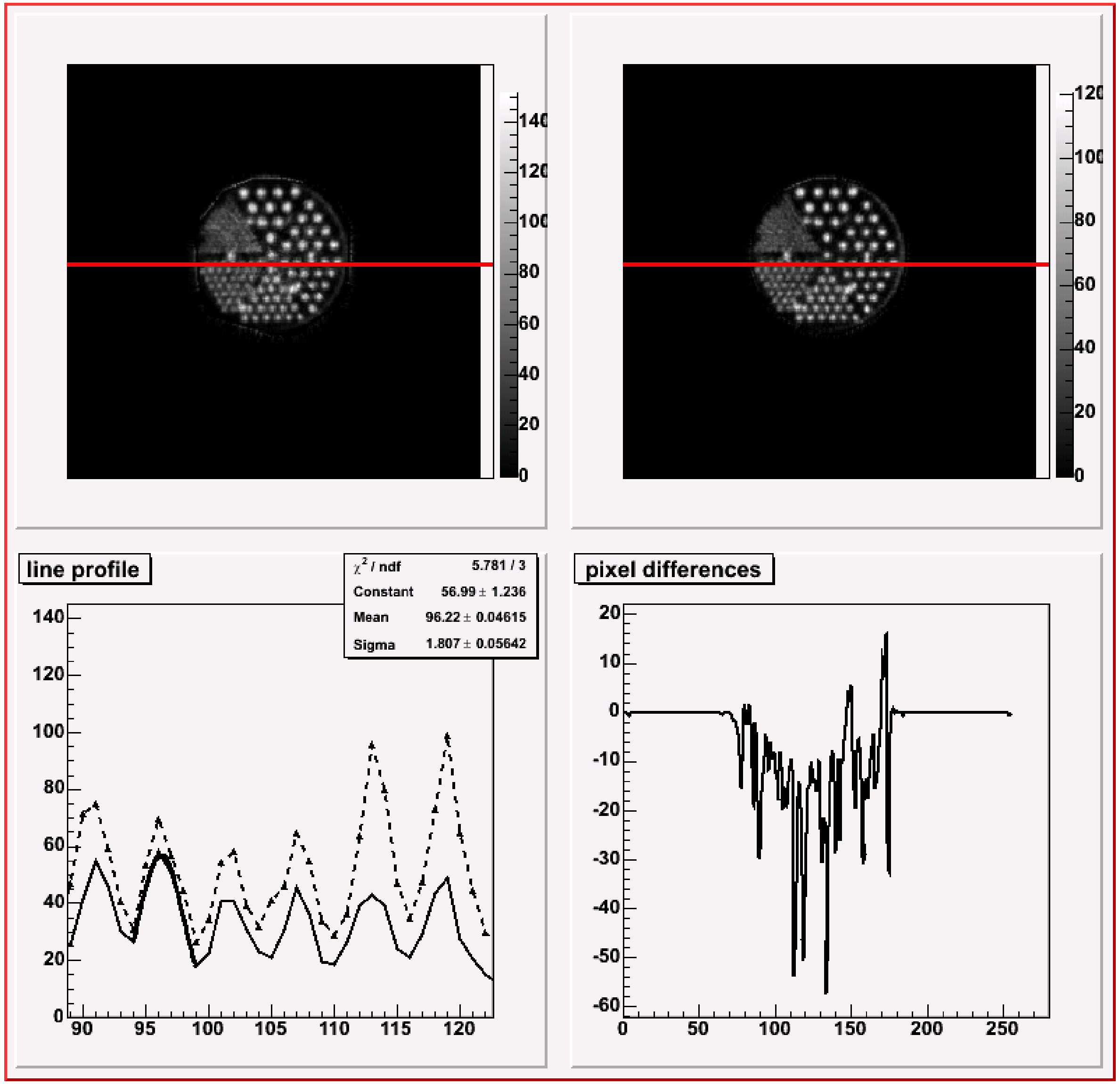
Resolution and Image Quality

- Comparison between the reconstrcuted images from the same rebinned data set using the OSMAPOSL executable with 18 resp. with 9 subsets.

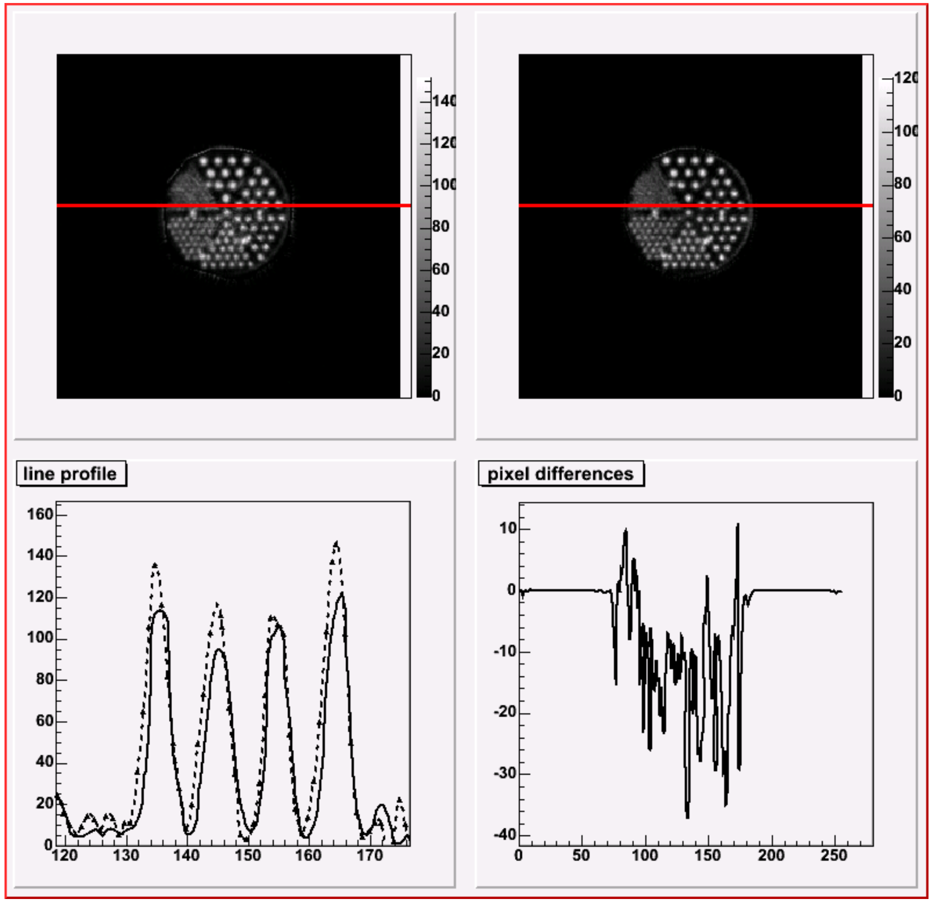


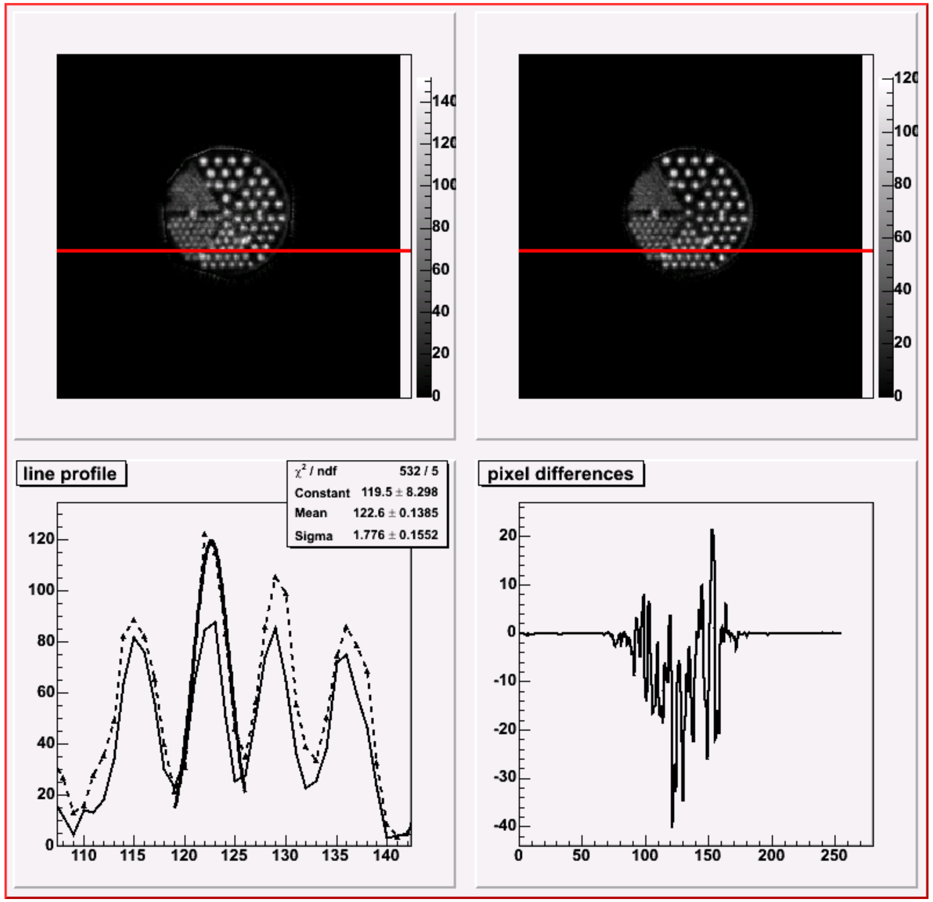
Resolution and Image Quality

- Comparison between the images reconstructed from the 3-D data set using OSMAPOSL and the 2D rebinned data set using the standard parameter set.
- The left image is derived from the rebinned data set, the right image from direct 3-D reconstruction.

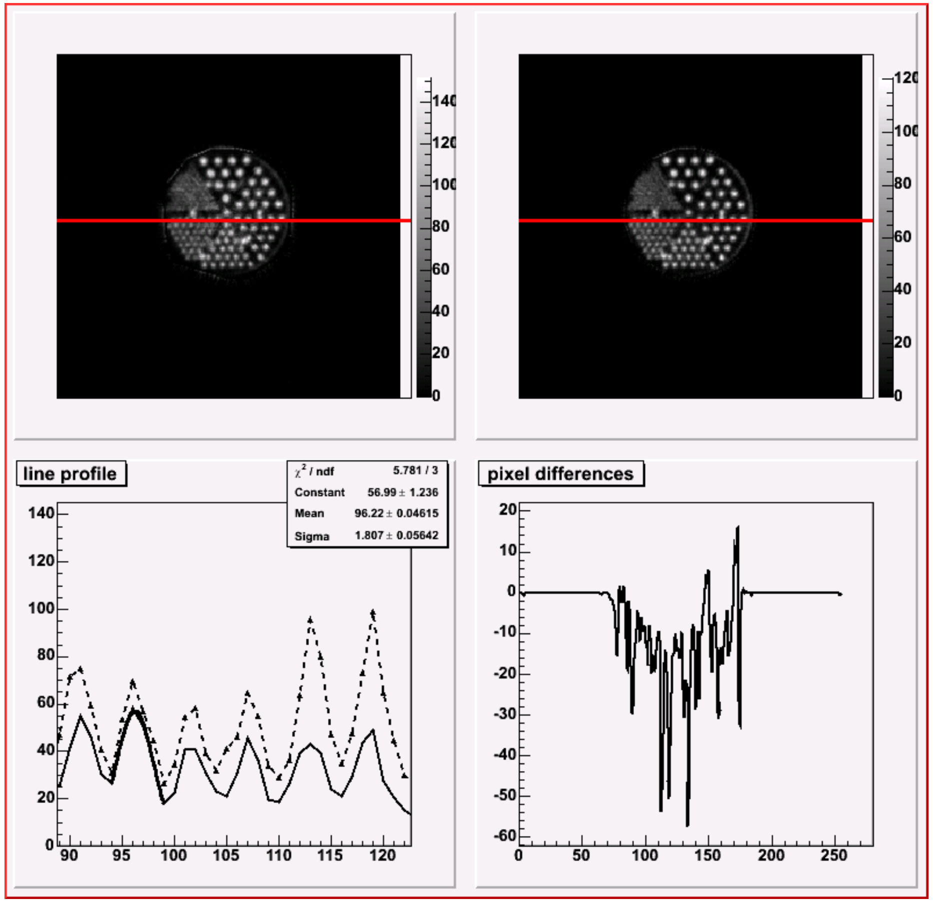


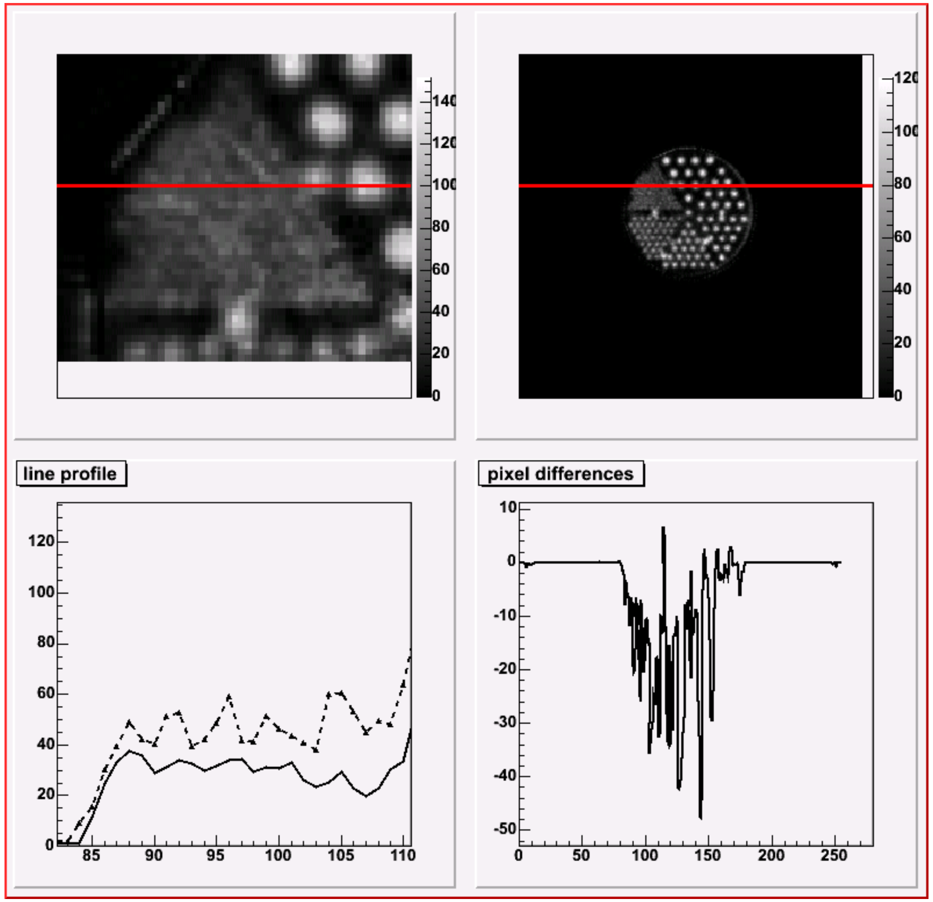
Resolution and Image Quality





Resolution and Image Quality





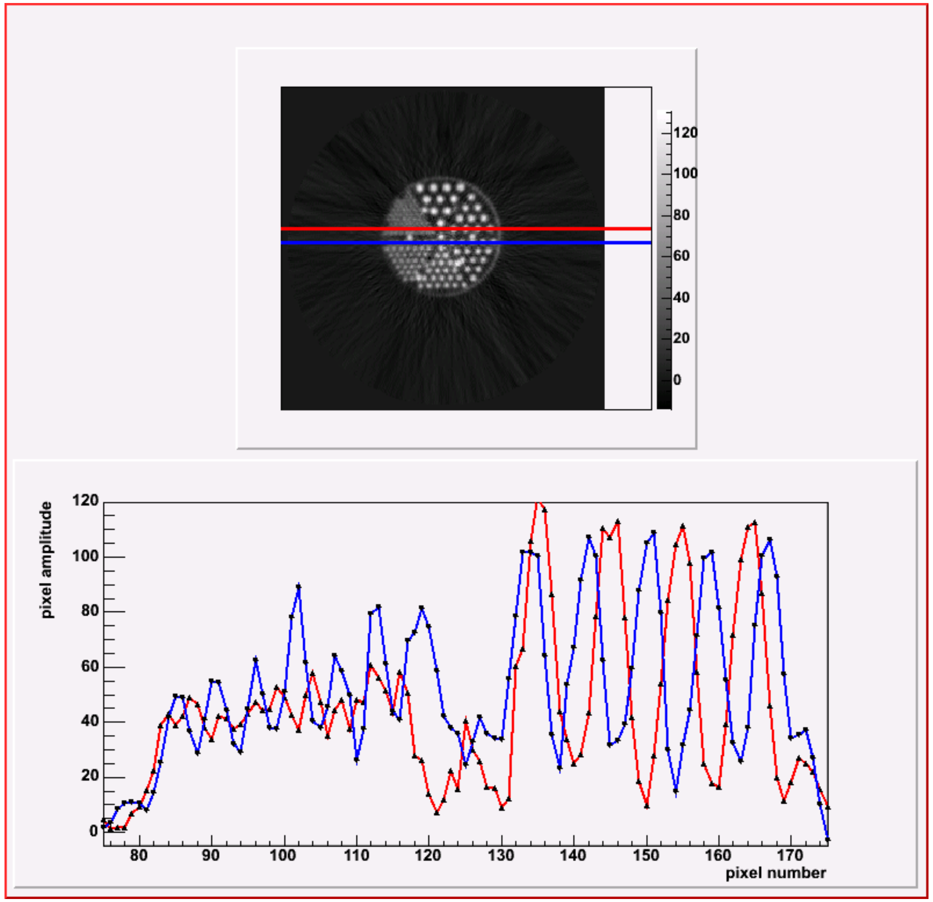
Resolution and Image Quality

- Effects of different parameter sets on image quality.
- The same 3-D data set as before was FORE rebinned using several different parameter settings.
- The FORE rebinned data set was reconstructed using filtered backprojection.

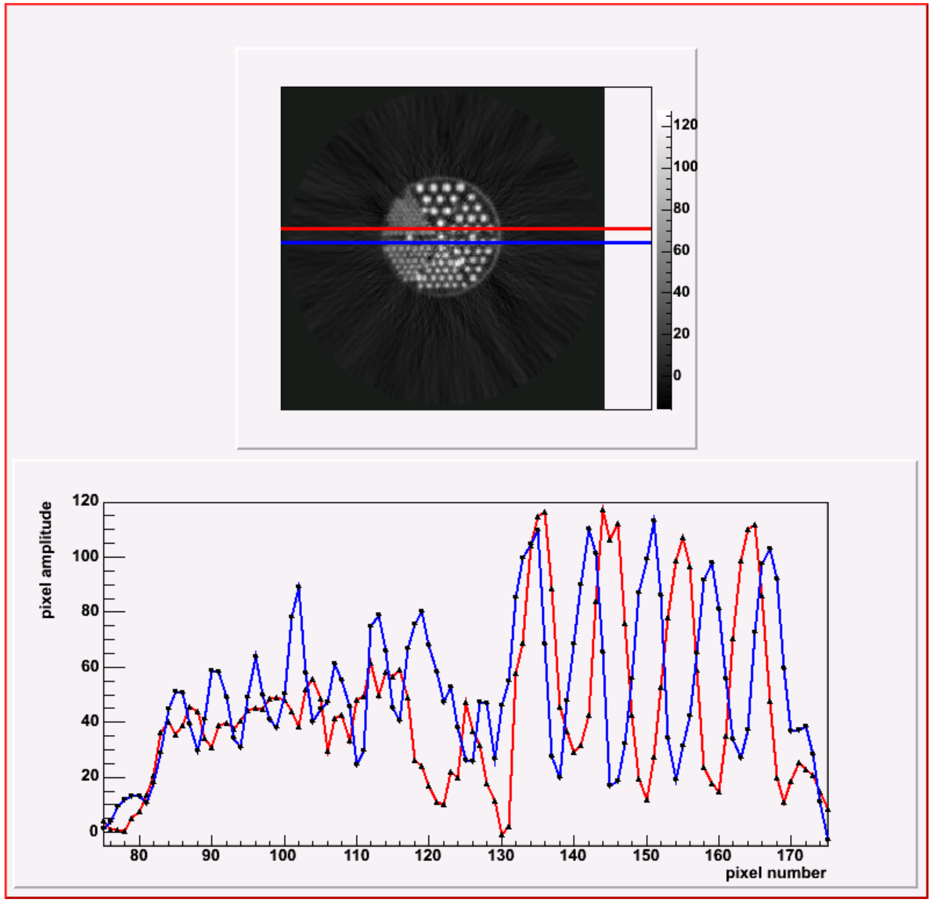
- Effects of suppressing the zeroth order term (SSRB) in the rebinning

Resolution and Image Quality

$\omega=2, k=2, kc=2, \delta=1$

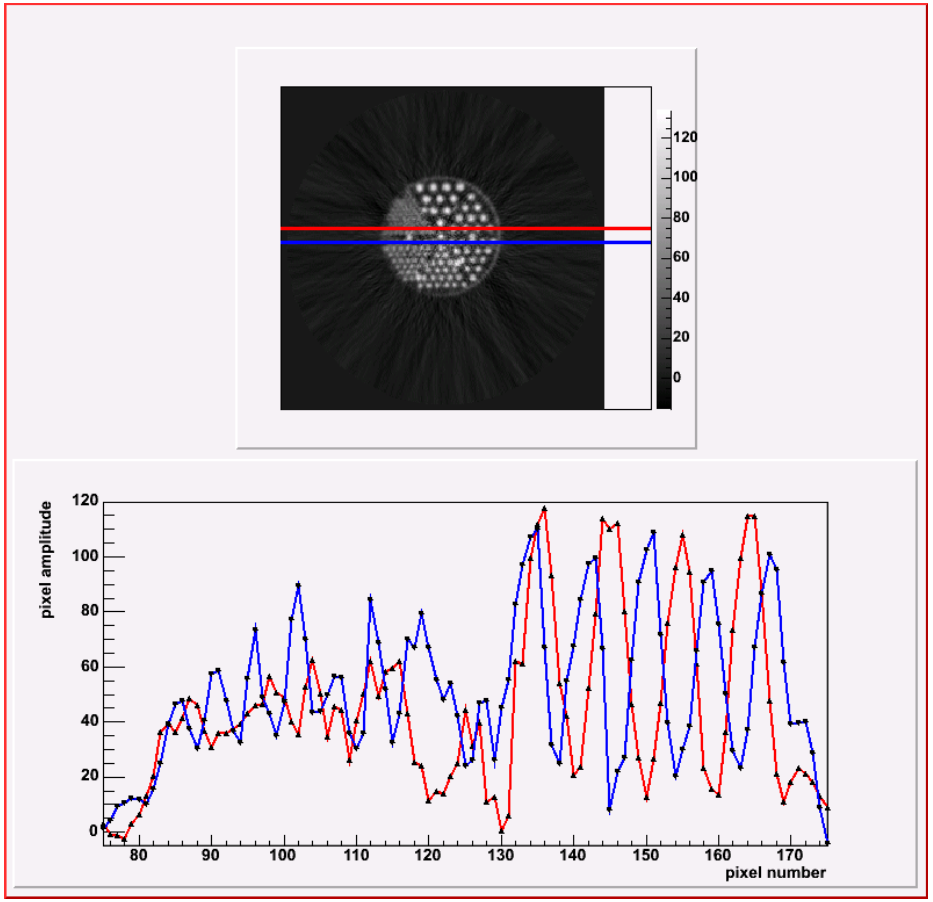


$\omega=20, k=20, kc=20, \delta=1$

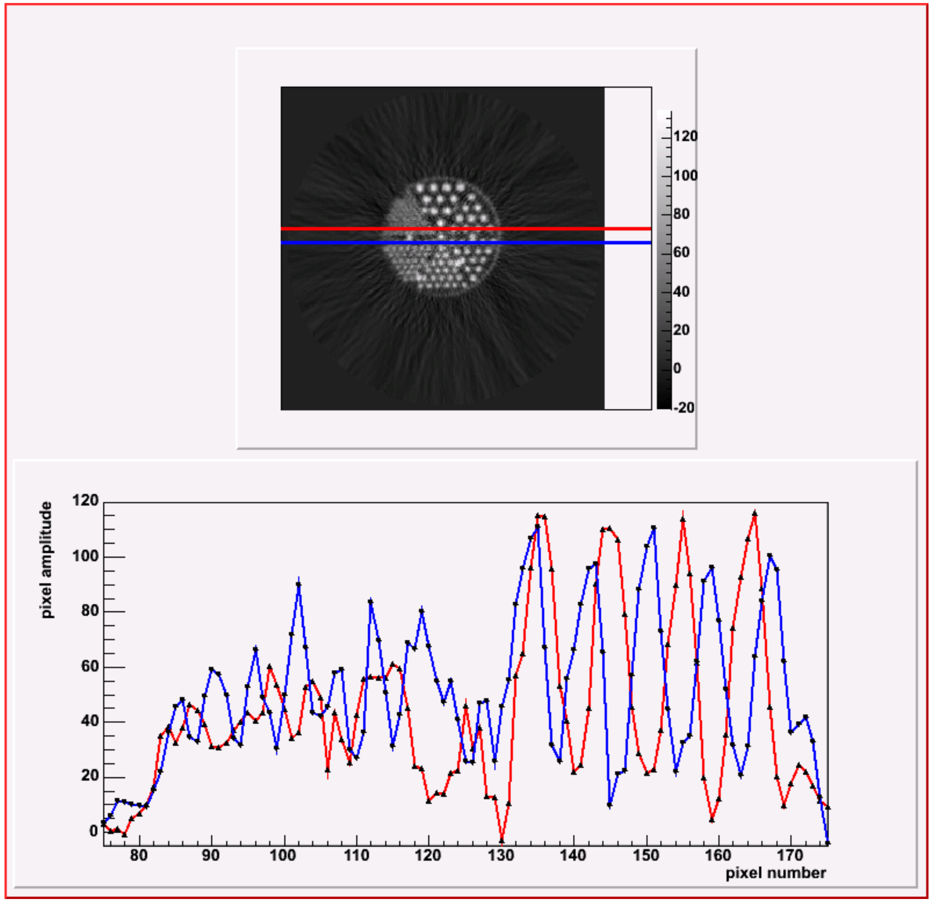


Resolution and Image Quality

$\omega=50, k=50, kc=50, \delta=1$

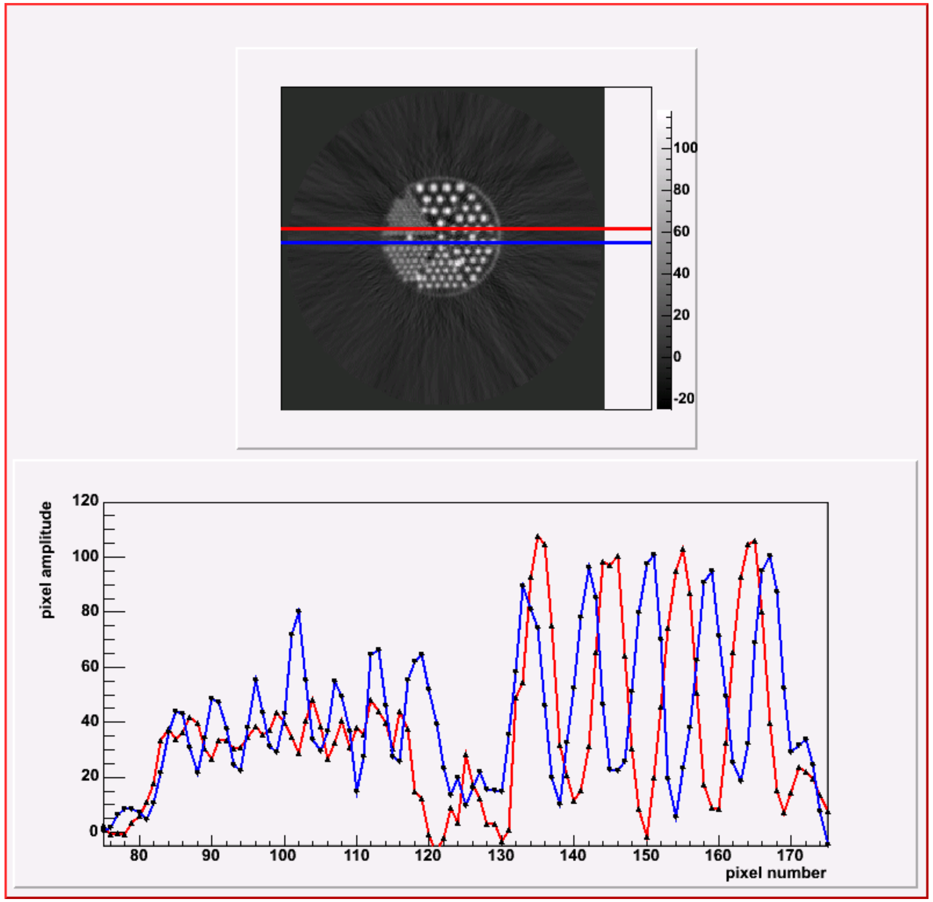


$\omega=100, k=100, kc=100, \delta=1$

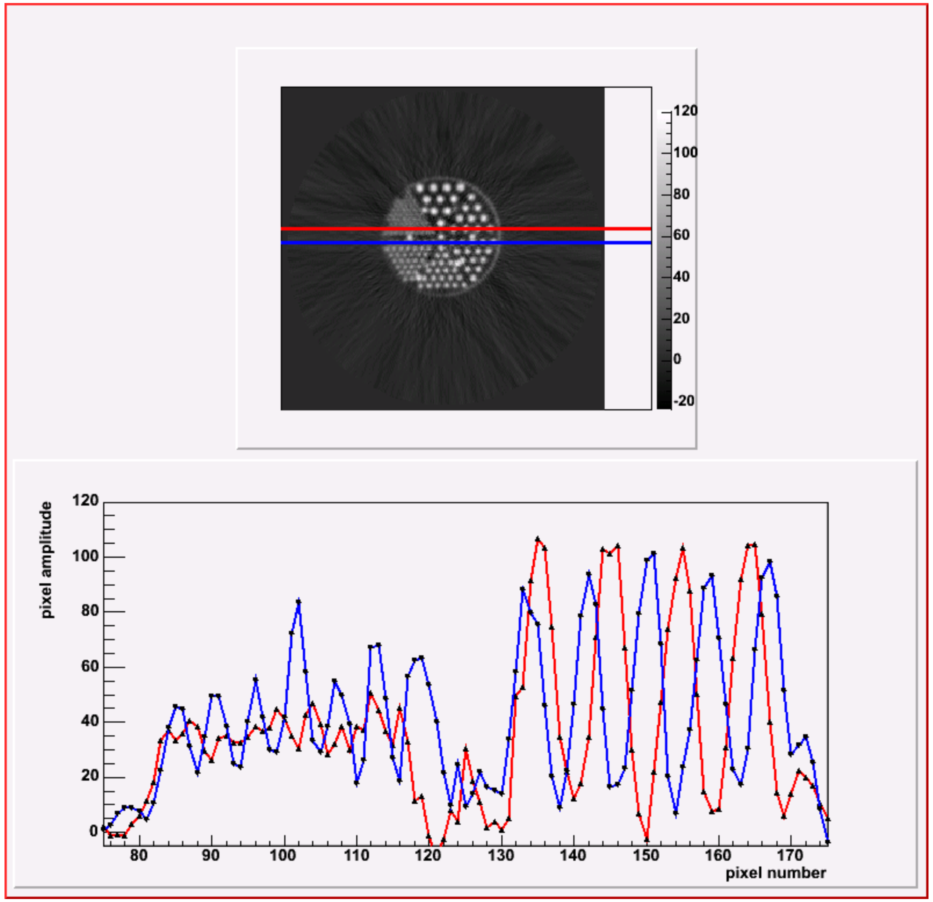


Resolution and Image Quality

$\omega=2, k=2, kc=2, \delta=30$

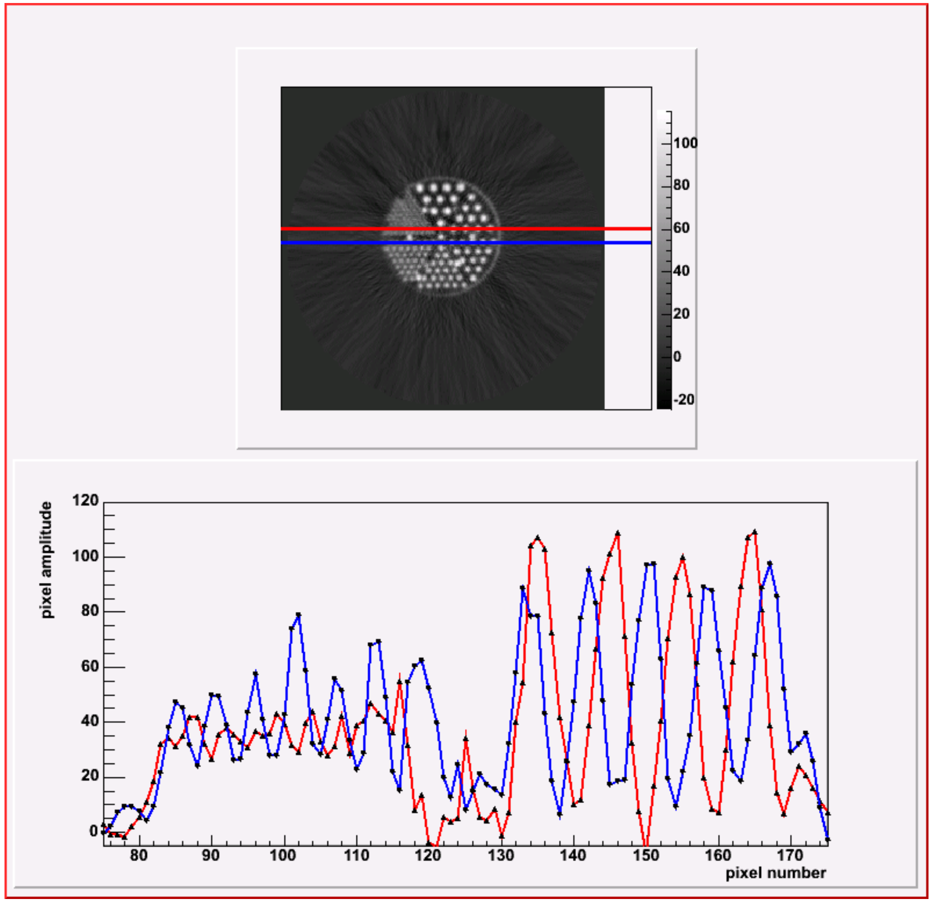


$\omega=20, k=20, kc=20, \delta=30$

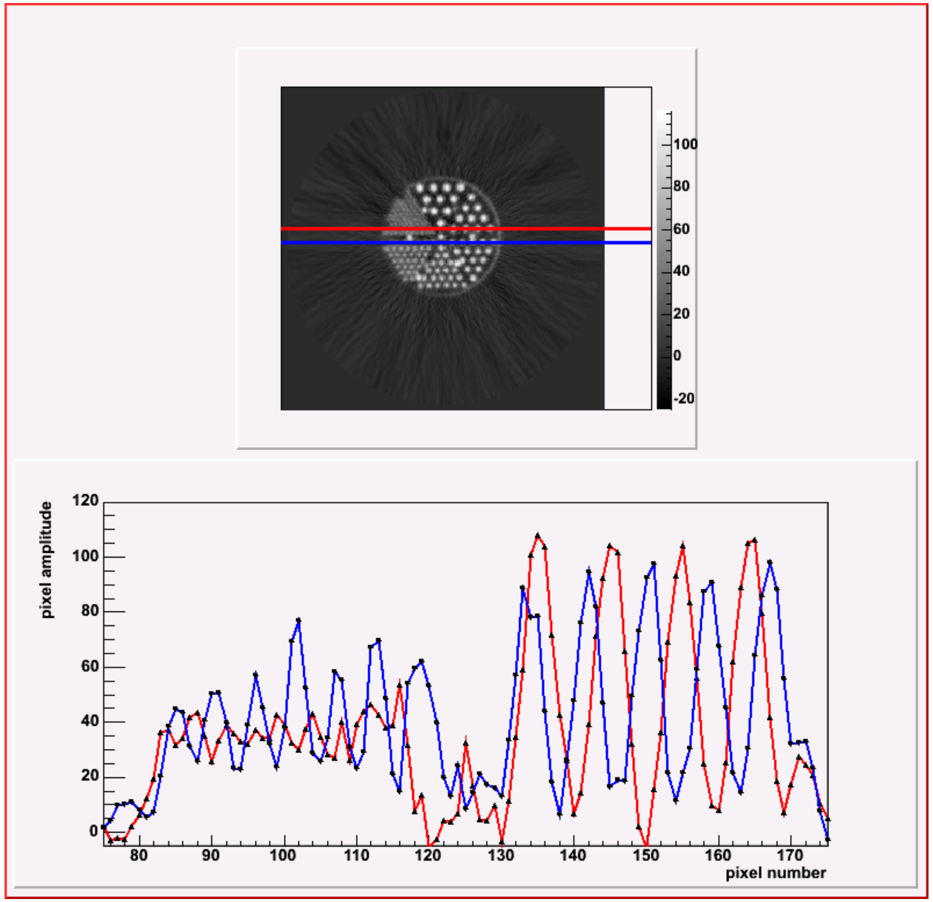


Resolution and Image Quality

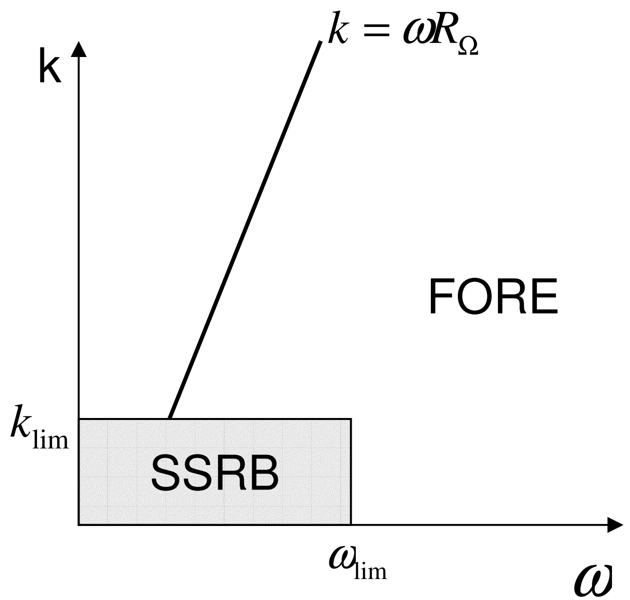
$\omega=50, k=50, kc=50, \delta=30$



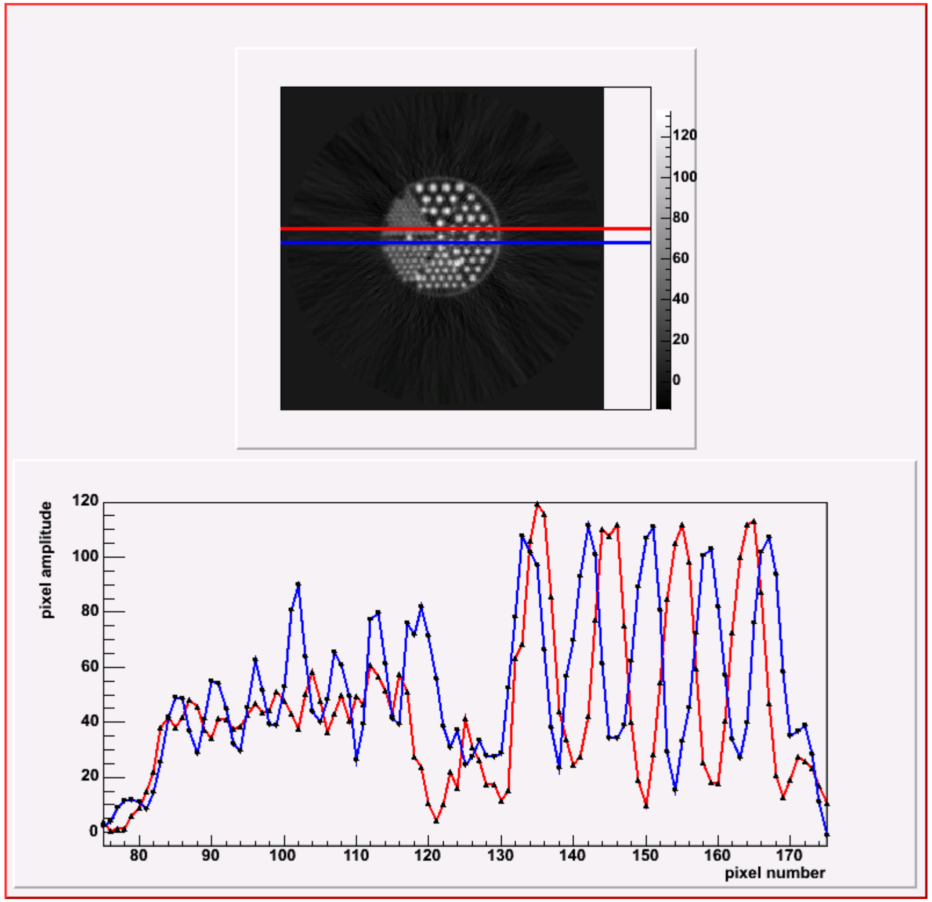
$\omega=100, k=100, kc=100, \delta=30$



Resolution and Image Quality



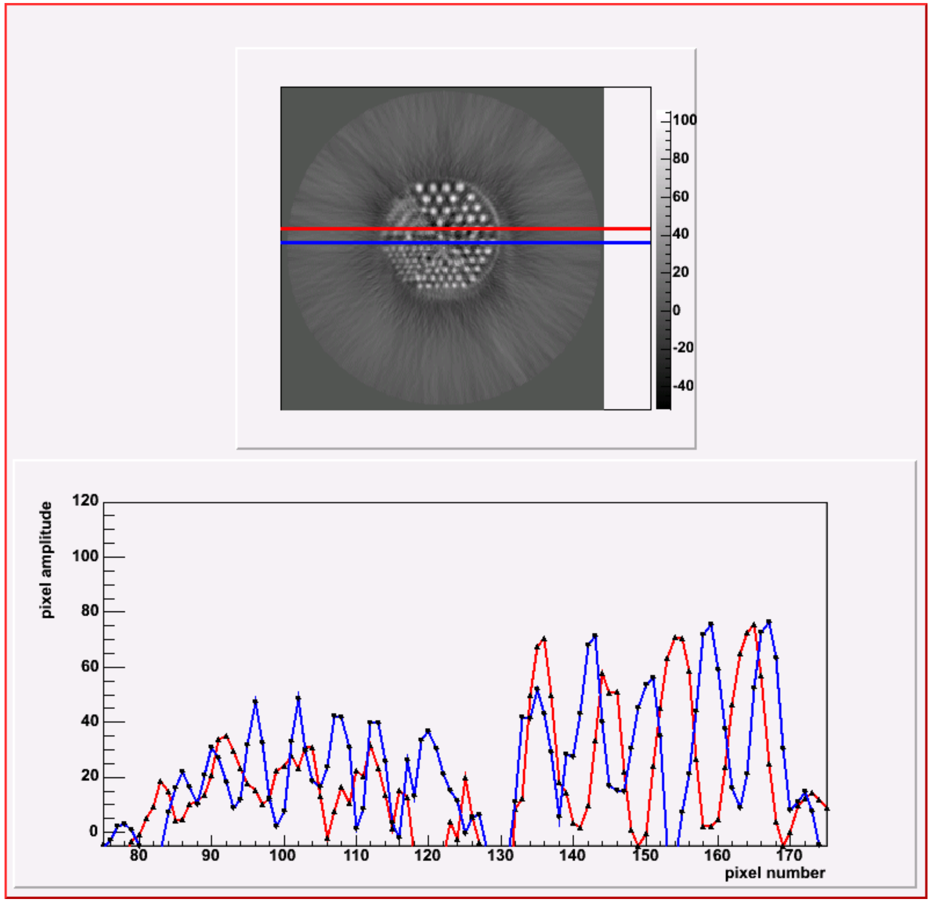
$\omega=2, k=2, kc=2, \delta=1$



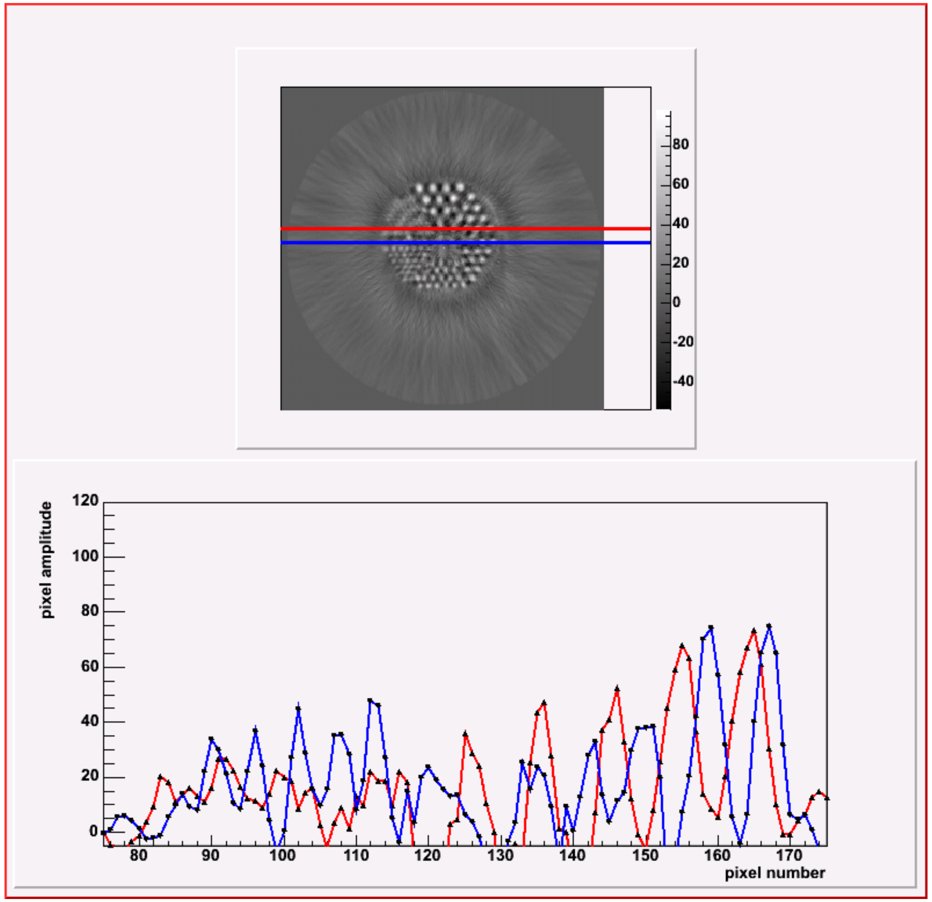
Effects of exclusion of the SSRB region.

Resolution and Image Quality

$\omega=10, k=10, kc=10, \delta=1$



$\omega=20, k=20, kc=20, \delta=1$



Conclusions

- The FORE algorithm has been introduced.
- The different computational steps performed during the FORE rebinning have been demonstrated and visualized for some line source data.
- A quick overview about implementation details and strategies was given.
- It has been shown how to call the FORE rebinning from your STIR based application and the necessary parameters which have to be provided in the parameter file were introduced.
- Profiling and time performance were discussed.
- Different FFT implementations, as the FFT is the most computing expensive part of the FORE rebinning, were discussed in terms of performance.
- Based on Jaszczack phantom data the resulting image qualities were discussed and comparisons between direct 3-D reconstruction algorithms and FORE rebinned 2-D reconstructions using OSEM and FBP were shown.