High Resolution Research Tomograph
USER COMMUNITY

HRRT_U 1.1 Specifications

1. General

HRRT_U 1.1 is an upgrade to previously released HRRT User Community Software releases 1.0 and 1.0.1. Its installation should not replace or modify any file of the original CPS distribution.

The main purpose of this version is to provide: Cologne specific gantry support, span 3 processing on 64-bit, a new Component-Based Normalization (CBN) with corrections for reported artifacts in previous evaluation versions, and an optimized 3DRP reconstruction.

- The previous program hrrt_rebinner_lut had no option to create the LUT for the Cologne specific gantry.

- Span3 processing on 64-bit: previous 32-bit versions of lmhistogram_u and norm_process didn't run on 64-bit in span 3 (not enough memory) and results of 64-bit versions were different from 32-bit versions. The programs are modified in the new release to use a LookUp Table (LUT) generated by a 32-bit program (hrrt_rebinner_lut.exe) to ensure the same results.

- Optimized 3DRP reconstruction to allow routine use in addition to OP-OSEM to verify if the reported bias in OP-OSEM is not significant.

- The normalization rod rotation is not perfectly centered and an option is added to CBN to use a rotation dwell fitted from data. The geometric factors are modified to correct for an efficiency drop for LORs with small incidence angle.

- Bug correction in lmhistogram_u and clc_u.

- Linux port of most command line programs.

- Time comparison between option in clc_u with millisecond precision.

2. Rebinner LUT: hrrt_rebinner_lut

A rebinner LookUp Table (LUT) is used to convert a Line Of Response in a sinogram address to guarantee the same result on different hardware and operating system.

The program hrrt_rebinner_lut is a windows 32-bit executable utility that should be run on a 32-bit windows computer to create the LUT file (it is recommended to use the hrrt_rebinner_lut.lut file included in the distribution). The program usage is hrrt_rebinner_lut -o lut_out_filename [-k] [-L mode], the options description is in table below:

Tag	Switch	Description	Req. Trace

S2.1	-o file	Specify output file	
S2.2	-k	Cologne gantry	
S2.3	-L [1|2]	Low resolution mode 1 or 2 : 1: binsize=2mm, nbins=160, nviews=144, 2:binsize=2.4375mm, nbins=128, nviews=144	

Table 1: hrrt_rebinner_lut options.

3. Histogramming: lmhistogram_u

3.1 Requirements trace

HRRT_U_1_1_requirements doc.

3.2 introduction

Lmhistogram_u creates sinogram(s) from 32-bit or 64-bit listmode data. The format is automatically determined from the file extension (.l32=32-bit, .l64=64-bit). 32-bit listmode data are generated by the hardware rebinner with a given span and maximum ring difference.
Lmhistogram_u creates multiple sinograms when frame durations are specified (dynamic study) or a single sinogram when no frame duration is specified (static study).
Lmhistogram_u creates 3D sinogram(s) for emission data and 2D for transmission/blank data.

3.3 Context

Lmhistogram reads the LOR to sinogram conversion LUT from the directory **c:\cps\users_sw** or **c:\cps\cluster_u** and transmission span from the gantry information file **gm328.ini** located in the directory specified by the environment variable **GMINI**.

Transmission data are histogrammed in 2D mode with a span specified by the tag **rebTxLUTMode0Span** in gm328.ini . The default value is 21.

Lmhistogram uses multiple threads:

1.A 64-bit read thread to load events into 64-bit event buffers (64-bit listmode input only).

2.A 32-bit read thread to load events into 32-bit event buffers (32-bit listmode input only).

3.A rebin thread to convert 64-bit event buffers into 32-bit event buffers (64-bit listmode input only).

4. A histogram thread to sort 32-bit event buffers into the sinogram .

5.A writer thread to output 32-bit event buffers when requested.

3.3 User interface

lmhistogram is a console command. The usage is:
lmhistogram lm_file –o sino_file [options]
lm_file is 64-bit or 32-bit listmode file, lm_file.hdr is the header file.
Options are described in table 2.

Tag	Switch	Description	Default value	64-bit list-mode only	Req. Trace
S3.1	NA	Lmhistogram_u uses a Rebinner LUT (c:\cps\users_sw\hrrt_rebinner.lut) and produces the same results on both 32-bit and 64-bit OS.	NA	yes	R2.1
S3.2	-o file.s	Creates sinogram file file.s, header file file.s.hdr and coincidence histogram file file.ch. Creates multiple sinogram frames files file_frameX.s, headers file_frameX.s.hdr and coincidence histogram files file.ch when specified in the listmode header.	Listmode filename with .s extension	no	
S3.3	-span X	Emission span, valid values: 3, 9. Transmission span is specified in GM328.ini by **rebTxLUTMode0Span** key. Listmode data type is specified in the listmode header with the keyword **!PET data type**. If the listmode header doesn't exist, emission data type is assumed.	9	yes	
S3.4	-rd	Emission maximum ring difference 2D sinogram only is created for transmission data.	67	yes	
S3.5	-PR -span 9	Creates span 9 prompts and trues sinograms for OP-OSEM reconstruction. Smoothed randoms span 9 sinogram will be created from the .ch file. The trues sinogram is used for scatter correction.	Trues span 9	yes	
S3.6	-PR -span 3	Creates span 3 prompts sinogram and span 9 true sinogram for OP-OSEM reconstruction. Smoothed randoms span 3 sinogram will be created from the .ch file. The scatter correction is computed in span 9 from and expanded to span 3 during reconstruction.	Trues span 9	yes	
S3.7	-d duration [,skip]	frame duration in seconds, optional skip (before) in seconds	NA	no	
S3.8	-d duration [*repeat]	Frame duration in seconds, optionaly repeat duration	NA	no	

S3.9	-scan	Reads the listmode file **file.l64** and creates a headcurve file **file_lm.hc** with events in the listmode. When the countrate is too high, data will be lost and the file will be different from the file **file.l64.hc** created during the acquisition with events reported by the coincidence Controller.	NA	yes	
S3.10	-count N	Stop histogramming after N true events.	NA	no	
S3.11	-add file.s	Add to existing sinogram file (only in true mode)	NA	no	
S3.12	-o file.l32	Create 32-bit listmode file.l32 and header file.l32.hdr	NA	yes	
S3.13	-mock file.s	Output shifted-mock correction sinogram for when histogramming a transmission file.	NA	Yes, TX only	
S3.14	-L type	Specify the low resolution mode. e.g -L 1 for Low Resolution mode 1	N/A	yes	
S3.15	-start [trues count]	Skips to the specified trues count rate and starts the histogramming from there.	0	no	

Table 2: lmhistogram_u options

Multiple -d specifiers indicate dynamic sequencing and will generate multiple sinograms and headers

e.g. lmhistogram file.l64 -span 9 -d 300,10 -d 300*11 -d 600*4

will generate 16 span 9 sinograms skipping the first 10 seconds

- 12 of 300seconds (test_frame0.s ... test_frame11.s)

- 4 of 600seconds (test_frame12.s ... test_fram15.s)

In 32-bit mode, an event contains the sinogram address and a bit indicating that the event is a prompt or delayed.

In 64-bit mode, an event is a Line Of Response LOR specified by the head-pair, (X, Y, DoI) of the 2 crystals; and a bit indicating that the event is a prompt or delayed. lmhistogram_u computes the sinogram address using the LUT.

In Trues mode, lmhistogram_u increments/decrements the sinogram address if the event is a prompt/delayed.

In separated prompts and delayed mode, lmhistogram_u increments the prompt sinogram address if the event is a prompt and delayed sinogram address(span 9) otherwise. In span 3 mode, the prompt sinogram is written to disk, then converted to span 9 and the delayed span 9 sinogram is substracted to create and write the span 9 true sinogram.

When the “-add file.s” option is specified, the sinogram is initialized with the specified file prior to the listmode event processing.

4. Scatter correction : e7_sino_u

The new e7_sino_u requires less memory (1.3Gb) than available (2Gb) on common 32-bit windows XP computers, as shown in table 2. The original e7_sino required more than 2GB memory and started swapping when loading the normalization. Its speed depended heavily on the disk speed, whereas there is no swapping in the new version. e7_sino_u deletes the normalization after using it and does not store the 3D scatter sinogram, which can be calculated on the fly from the 2D scatter to 3D span 3 or span 9 using Inverse Single Slice Rebinning (ISSRB) and scale factors during reconstruction.
A new option -os2d allows to save the scatter in a new compact format (2D segment followed by the scale factors). The scatter is always calculated in span9 (see also lmhistogram_u). The new scatter format is extended on the fly to span9 or span3 by hrrt_osem3d.

OLD: e7_sino (2.5GB)

Action	Required Memory
Load EM	636MB
Load ATN	1.2GB
Load Norm	1.8GB
EM = gapfill(EM*Norm)	
im = recon(SSRB(EM))	1.85GB
mu = recon(ATN)	
s2D = scatter_sim(im,mu)	1.9GB
s3D = ISSRB(s2D)	2.5GB
SF= ScaleFactors(ATN,EM,s3D)	
Save (SF*S3D)	

NEW: e7_sino_u (1.3GB)

Action	Required Memory
Load EM	636MB
Load Norm	1.2GB
EM = gapfill(EM*Norm)	
Delete Norm	636MB
Load ATN	1.2GB
Im = recon(SSRB(EM))	1.25GB
mu = recon(ATN)	
s2D = scatter_sim(im,mu)	1.3GB
SF=ScaleFactors(ATN,EM,ISSRB(s2D))	
Save (SF*ISSRB(s2D)) or Save s2D and SF	

Table 3.1: e7_sino and e7_sino_u memory usage.

4.1 Modified Functions

None.

Tag	Switch	Description
S3.0	Backward compatibility	E7_sino_u should give the same results (with 0.1% difference) that e7_sino when the same inputs and arguments are used.

Table 3.2: Tags and descriptions of the modified functions in e7_sino_u.

4.2 New Functions

Tag	Switch	Description
S3.1	--os2d	New option to output the scatter in compact format: unscaled 2D segment followed by the scale factors. The switch has no argument; the output file

		name is specified using the same switch (-os) as in e7_sino.
S3.2	--lebr b	New option to specify back and front crystal layer background energy ratio (LBER) that overrides values from GM328.ini. Example: –lber 12.0.
S3.3	-m 3281	Low Resolution mode 1: 2mm bin size, 2.4375mm plane separation
S3.4	-m 3282	Low Resolution mode 2: 2.4375mm bin size and plane separation

Table 3.3: Tags and descriptions of the new functions in e7_sino_u.

5. Fast Inki Reconstruction: hrrt_osem3d

Tag	Switch	Description
S5.1	-W 3	Reconstruction using hrrt_osem3d should give the same image as the standard cluster reconstruction using the same input files and the OP-OSEM scheme.
S5.2	-B 0,0,0	hrrt_osem3d supports resolution recovery (PSF) as described in the paper [1].
S5.3	-s	hrrt_osem3d supports a new scatter format (2D segment followed by scale factors) described in S3.1. It automatically detects the format (no new switch) and extends the 2D scatter into 3D span9 or span3 using Inverse Single Slice Rebinning (ISSRB) and applying the scale factors.
S5.4	-d	hrrt_osem3d supports coincidence histogram (.ch) as input for smoothed randoms. It automatically detects the format (no new switch) and computes the smoothed randoms using Inki's optimized gen_delays from the .ch file and saves it in a temporary file for 32-bit OS (2GB) and stores it in memory for 64-bit OS (4GB RAM or more) only.
S5.5	-W 2	hrrt_osem3d supports ANW schemes (Attenuation Normalization Weighted) to reconstruct data without attenuation (e.g. point source) as supported by the original cluster reconstruction code. The missing attenuation is replaced by 1 and scatter by 0.
S5.6	-M 328	HRRT scanner model (same as –M 999)
S5.7	-M 3281	Low Resolution mode 1: 2mm bin size, 2.4375mm plane separation, 103 planes.
S5.7	-M 3282	Low Resolution mode2: 2.4375mm bin size and plane separation, 103 planes.
S5.8	-D dir	Temp directory to store normfac and smoothed randoms (e.g patient directory) for 32-bit OS. The smoothed random sinogram created from .ch file is deleted at the end of the reconstruction.
S5.9	N/A	Use a LookUp Table (LUT) for Nearest Neighbor(NN) rebinning to avoid rounding mismatch between random smoothed sinogram that may be created with 64-bit executable (hrrt_osem3d) and the normalization created with 32-bit executable. The LUT file **hrrt_rebinner.lut** is stored in **c:\cps\users_sw** on the reconstruction server or **c:\cps\cluster_u** on the cluster reconstruction nodes.

S5.10	N/A	Write reconstruction software version and build ID in the image header.

Table 1: Tags and descriptions of the hrrt_osem3d specifications.

6. Cluster Integration: hrrt_osem3d

6.1. clc_u and clq_u

The new clc_u is a modified version of the original cluster service "ClusterServer" to support subclustering and different applications (clcrecon). Only the osem3d application was supported in original versions. The new service is installed using the same service name as the original programs. The user can easily switch from the new to the original programs using the command "clc -U" ("clc_u -U" to switch back to the new programs).

clq_u is a new program to install the queue service "ClusterQueue" which was not part of the original software.

The cluster configuration is stored in c:\cps\cluster_u\ClusterConfig.txt (example below).

```
HOST   reconserver
RAID   reconserver     192.168.1.1
node   node1                                   192.168.1.11
.
.

node   node8                                   192.168.1.18
.
```

The file contains two entries for the server (one to specify that it is the server and one for its IP address) and one for each node (IP address).

The Queue configuration is stored in c:\cps\cluster_u\QueueConfig.txt (example below for an 8 node cluster).

```
#!QueueConfig
Recon-Jobs      D:\Recon-Jobs
Jobs-Submitted D:\Recon-Jobs\Jobs-Submitted
Jobs-In-Progress        D:\Recon-Jobs\Jobs-In-Progress
Jobs-Completed          D:\Recon-Jobs\Jobs-Completed

ClusterSize.Min         1
ClusterSize.Max         8
ClusterSize.Set  1
```

ClusterSize.Max is the number of nodes. ClusterSize.Min and ClusterSize.Set are set to 1 to force the sub-cluster size to 1. Such a cluster can reconstruct eight frames in parallel.

6.1.1 clc_u functions

The functions in table 4 below is used to install clc_u or clq_u and test the services. Refer to keywords

document for a complete list of functions

Tag	Switch	Description
S6.1	-v	Print clc/clq version.
S6.2	-I	Install ClusterServer/QueueServer service on the local machine
S6.3	-U	Update ClusterServer/QueueServer on all machines (server and nodes)
S6.4	-V	Print ClusterServer/QueueServer versions
S6.5	-P [computer name]	Read the time incl. milliseconds off the computer issuing the command and the computer given as an argument.

Table 4: Tags and descriptions for cluster integration.

6.2 clcrecon

clcrecon is a modified version of the existing osem3 cluster application to run hrrt_osem3d on a single cluster node. The cluster must be configured in subclusters with a single node per subcluster. If a subcluster has many nodes, only the first node will be used). The application uses two threads: one to execute hrrt_osem3d and one to monitor the reconstruction progress. clcrecon replaces input file paths by UNC paths prior to executing hrrt_osem3d (e.g.: D:\SCS_SCANS is replaced by \\SERVER\SCS_SCANS).

6.2.1 Modified Keywords

Tag	Switch	Description
S6.5	#!clcrecon	First line in job file (instead of #!osem3d).
S6.6	Scatter	3D or new (unscaled 2D and scale factors) scatter file.
S6.7	Delayed	Smoothed random or coincidence histogram (.ch).
S6.8	N/A	Input and output file names are replaced by UNC paths

Table 5: Tags and descriptions of osem3 modified keywords.

Refer to Cluster Keywords 28-JUL-2008 document for complete list of keywords.

6.3 ClusterServer account configuration

The reconstruction using clcrecon and hrrt_osem3d is initiated by ClusterServer service. The Service processing account should be set on all nodes to an existing account on the fileserver with read/write access to the data directory D:\SCS_SCANS. See installation procedures for details.

7. Component-Based Normalization (CBN): norm_process

norm_process is an implementation of the CBN method developed by NIH. It computes the crystal efficiencies and a normalization sinogram from the rotating ^{68}Ge rod normalization scan listmode. It requires a much shorter acquisition time (e.g. 1h) than the original direct normalization method (e.g. 32h) when using a new rod.

The normalization is computed in 2 steps: (i) crystal efficiency, (ii) normalization sinogram in specified span (span 9, span 3, low resolution 2.4mm or 2mm). The default is span9.

The crystal efficiencies are saved to be reused to compute the normalization sinogram of (other) spans.

The input file can be either a 64-bit listmode (.l64 extension) or a crystal efficiency file (.ce extension).

If the input is a listmode, the outputs are crystal efficiencies and a normalization sinogram in the requested span (default=9).

If the input is a crystals efficiency, the output is a normalization sinogram in the requested span (default=9).

7.1 Algorithm

1) Computes the crystal fansums, applying the corrections: (i) rotation dwell, (ii) radial and axial geometric effects, (iii) solid angle; for each LOR.

2) Computes the crystal efficiencies iteratively from the fansum and saves the crystal efficiencies (.ce file extension).

3) Computes the normalization sinogram in requested span from the crystal efficiencies and saves it.

The original CBN algorithm assumed a perfectly aligned rotation with the center of the scanner FOV as the center of rotation. Tests on different scanners showed that the rotation was not centered and that the correction for the misalignement was required. The new algorithm includes a preliminary step to compute the rotation dwell from the data using the standard direct normalization (compute_norm) method; and uses the computed dwell instead of the perfect dwell.

7.2 Geometric Profiles

The axial and radial geometric profiles in the original software from NIH are stored in a text file c:\cps\users_sw\gr_ga.txt as functions of the LOR incidence angles with the face of each crystal. The file contains discretized values: one for every five degrees for the front and back layers respectively.

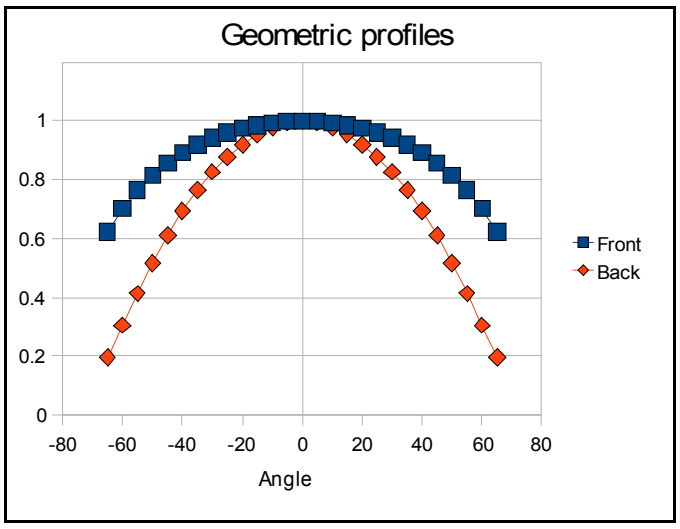

Figure 1: original geometric profiles

It was found in [2] that the LORs with small incidence angle had an efficiency drop of about 10% compared to neighboring ones. The geometric profiles were modified as shown on figure 2 to correct for this effect. The new geometric profiles are specified as 4 t-uples (breakpoint, a0, b0, a1,b1) as shown on table below:

Break point	**Front Layer slope, intercept**	**Back Layer slope, intercept**	**Range**
0	-2.26, 15.0	-3.26, 22.2	$0 \le \theta < 5$
5	0.62, 0.82	1.07, 1.83	$5 \le \theta < 22.5$
22.5	0.72, -2.08	1.19, -1.12	$22.5 \le \theta < 45$
45	1.08, -18.85	1.32, -6.92	$45 \le \theta$

Table 6: geometric profile parameters $g(\theta) = \cos(a(\theta)\cdot\theta + b(\theta))$

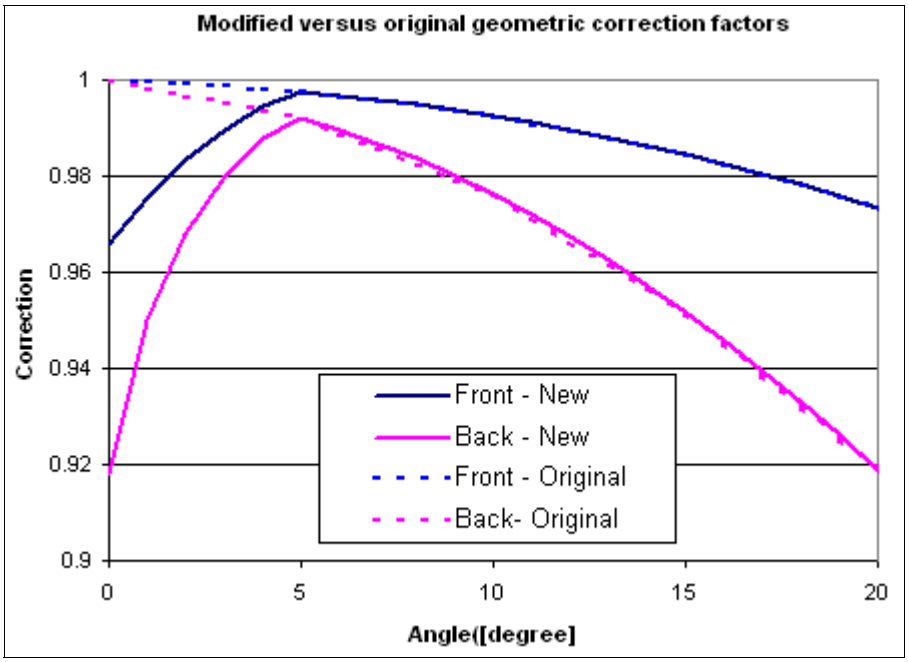

Figure 2: Modified versus original geometric profiles, symmetric part for negative angles not shown

7.3 Functions

norm_process is a command line function. The usage is: norm_process lm_file|ce_file [-o norm_file] [options].

The command prints the usage if no argument is provided. The first argument is the input file (listmode or crystal efficiencies). If the output file name is not specified with -o option, it is derived from the input file name by changing the extension to .n.

Tag	Switch	Description
S5.1	-s span,rd	Specify the span and optionally the maximum ring difference. The defaults are 9 and 67.
S5.2	-d duration	Only process the specified scan duration from the beginning of the listmode data (given in seconds).
S5.3	-L LUT	Specify rebinner LUT to use (standard, Cologne, Low Resolution 1 or 2)
S5.4	-g geo_fname	Specify radial and axial geometry filename, old or new format
S5.5	-I max_iter	Set maximum number of iterations (default=30)
S5.6	-M min_rmse	Set minimum RMSE for convergence criteria (default=10e-9)
S5.7	-c corr_bits	Set corrections to be applied when computing fansum: Obliqueness=0x1, RotationDwell=0x2, SolidAngle=0x4, GeomRadial=0x8, GeomAxial=0x10. All corrections are applied by default. Example: -c 7 to exclude geometric radial and axial corrections.
S5.8	-R dwell_sino	Rotation dwell sonogram

Table 6: Tags and descriptions of norm_process function.

8. Transmission scatter correction : e7_atten_u:

Modified Functions

None.

Tag	Switch	Description
S8.0	Backward compatibility	E7_atten_u should give the same results (with 0.1% difference) that e7_atten version when the same inputs and arguments are used.

New Functions

Tag	Switch	Description
S8.1	--txsc a,b	Specifiy transmission scatter correction a,b factors: ln(ACF) = a + b*ln(bl/tx) Default: 0,1.0 is equivalent to original method without scatter correction. The scatter is accounted for by scaling the mu-map so that the histogram peak (water peak) is 0.096. Recommended values: -0.1,1.18 (see TX-Scatter-correction-21jul08.pdf). When b is not one, an absolute scaling (0.096/0.086) is applied to get mu value at 511KeV from 662KeV.
S8.2	--txblr r	Specify TX/BL ratio instead of getting it from data (under the bed sinogram

		bins).

9. Reconstruction Graphical User Interface: ReconGUI_u

ReconGUI_u is a modified version to use the new versions (lmhistogram_u, e7_atten_u, e7_fwd_u, e7_sino_u, hrrt_osem3d).

Modified Functions

Tag	Switch	Description
S9.1	N/A	Supports 3D OSEM reconstruction without a cluster using hrrt_osem3d
S9.2	N/A	Static reconstruction jobs are always executed on the file server
S9.3	N/A	Dynamic reconstruction jobs are queued to the cluster if enabled (cluster submit directory not empty).

New Functions

Tag	Keyword in recon.ini	Description
S9.4	Use PSF (0=NO, 1=YES)	New keyword in recon.ini to enable/disable using PSF. Default is disabled. ReconGUI_u has check box showing the status in the reconstruction configuration status. Ex: Use PSF (0=NO, 1=YES) := 1
S9.5	Mu Zoom	New keyword in recon.ini to specify sinogram transaxial and axial rebinning for MAP-TR mu-map reconstruction. Example with axial rebinning: Mu Zoom: = 2,3 Example with no axial rebinning: Mu Zoom: = 2,1
S9.6	TX Scatter Factors	New keyword for transmission scatter correction (see e7_atten in section 10). Ex.: TX Scatter Factors:= -0.1,1.18
S9.7	TX_TV3DReg	Integrate Total Variation 3D Regularization (TV3DReg) as an alternative to MAP-TR segmentation in recon.ini. Example: 1- Head=4,0.0,0.005,0.03,10.,0.096,0.02,0.11,10.,0.03,0.07,0.105;$ 2-Germanium phantom=2,0.000,0.005,0.103,0.005,0.050;$ 3-Water phantom=2,0.000,0.005,0.096,0.050,0.050;$ 4-TX_TV3DReg=0.5,2,3

		When TX_TV3DReg is selected, ReconGUI_u calls e7_atten_u without segmentation and calls TX_TV3DReg to apply TV3DReg on the mu-map image created by e7_atten_u.
S9.8	N/A	Add segmentation method (MAP-TR or TX_TV3DReg) to the mu-map image header.
S9.9	N/A	Ignore TX scatter correction when using MAP-TR for backward compatibility.
S9.10	Txblr	Get TX/Blank ratio from TX and BL headers and call e7_atten_u with the ratio instead of getting the ratio from sinograms (under the bed sinogram bins).
S9.11	LBER	Extract LBER from norm header if present and call e7_sino_u with the value. Ex.: LBER := 12.0
S9.12	N/A	Create e7_atten_u, e7_fwd_u, e7_sino_u and hrrt_osem3d log files in the log directory under the patient directory.
S9.13	N/A	Create e7_atten_u and e7_sino_u QC files in the QC directory under the patient directory.
S9.14	EM HIST start	Skips to the specified trues count rate and starts the histogramming from there.

10. Interfile to ECAT Conversion: if2e7

if2e7 2.3.0 is a merge of features from:
–Amsterdam (date manipulations, -q option to remove hot pixels)
–Aarhus to enter patient weight and dosage information (strength and time)
–John Hopkins to enable scanner dependent dead-time

Modified Functions

None

Tag	**Switch**	**Description**
S10.0	Backward compatibility	If2e7 should give the same results (with 0.1% difference) that previous version when the same inputs and arguments are used.
A10.1	-u units	Specify output image units. Valid units are Bq/ml, kBq/ml and Mbq/ml. Default is Bq/ml.

Table 6: Tags and descriptions of norm_process functions.

New Functions

Tag	Switch	Description
S10.1	-F name	Set Hospital (Facility) name
S10.2	-D dirname	Specify a secondary storage directory (the ECAT image is copied in a patient sub-directory)
S10.3	-T	Trim empty planes using boundaries extracted from the mu-map
S10.4	-S dirname	Specify calibration files directory. If2e7 searches the directory for calibration files and selects the one with matching scan date.
S10.5	N/A	Use mu-map mask to find image extrema and avoid hot spots

11. Scatter Parameter Adjustment Tool: calcringroiratio

calcringroiratio 1.0.0 is a recompiled version of Roman Krais's original version 0.3.0.

New Functions

None.

Modified Functions

None.

12. Calibration:CorrectRun_u

CorrectRun is a GUI for determining the calibration factor from a uniform cylinder image.

Bug Fix

CorrectRun applies inter-frame decay and computes correct calibration factors using ^{18}F or ^{11}C.

New Functions

CorrectRun_u has a new parameter “deadtime constant” in the configuration file c:\cps\users_sw\CorrectRun.ini. The parameter allows to use site-dependent constant for dead time correction as that is done in Interfile to ECAT conversion using if2e7. For example, the line below is used at Rigshopitalet in Copenhagen:

deadtime constant := 7.3e-006

Modified Functions

CorrectRun applies inter-frame decay and computes correct calibration factors using ^{18}F or ^{11}C.

13. 3DRP (3D FBP) reconstruction: ecat_3drp

Inki has optimized a 3DRP (3D ReProjection of missing data) software from Claude Comtat to take under 1 min. per frame on a fast computer. The code was initially developed by Michel Defrise and Christian Michel and can save the image in ECAT7 or Interfile format depending on the file extension (.v for ECAT7 and .i for Interfile). The program input is a fully corrected sinogram (random subtraction, normalization, gap filling, attenuation and scatter subtraction). Two correction programs are provided: hrrt_gapfill and hrrt_sinocor.

13.1 hrrt_gapfill

hrrt_gapfill is a correction program provided by Floris Van Velden from Amsterdam with two types of gap filling: linear interpolation in the angular direction and Constrained Fourier Space (confsp). The program usage and options are described in the table below.

Tag	Switch	Description	Required
S13.1.1	-p file	Prompt sinogram filename	yes
S13.1.2	-d file	Smoothed delayed filename	yes
S13.1.3	-n file	Normalization filename	yes
S13.1.4	-o file	Output filename	yes
S13.1.5	-a file	Attenuation filename (default = no attenuation correction)	No
S13.1.6	-s file	Scatter file name (default = no scatter correction)	No
S13.1.7	-g number	Gap filling method (0=none, 1=linear interpolation, 2=confosp). The default method is confosp.	No
S13.1.8	-i number	Confosp number of iterations (default is 15).	No
S13.1.9	-v	Verbose mode	No

13.2 hrrt_sinocor

hrrt_sinocor is an implemention of Claude Comtat method using the reprojection of OSEM image for the gap filling. The program uses Inki's fast projection code. It supports .ch file as input for smoothed delayed and 2d scatter file. The program usage and options are described in the table below.

Tag	Switch	Description	Required
S13.2.1	-p file	Prompt sinogram filename	yes
S13.2.2	-d file	Smoothed delayed filename. The filename may be coincidence	yes

		histogram (.ch) or smoothed delayed (ra_smo.s)	
S13.2.3	-n file	Normalization filename	yes
S13.2.4	-i file	Input filename for gap filling (use e.g. OSEM reconstructed image).	yes
S13.2.5	-o file	Output filename	yes
S13.2.6	-a file	Attenuation filename (default = no attenuation correction)	no
S13.2.7	-s file	Scatter file name (default = no scatter correction). The file may be a 3D scatter sinogram or an unscaled 2D sinogram followed by scale factors.	no
S13.2.9	-v	Verbose mode	no

14. Acquisition: scanit_u

New Functions

None.

Modified Functions

- Milliseconds are added to the can start time in the listmode header for motion and listmode files synchronization, example:
  !study time (hh:mm:ss.sss) := 15:40:26.915
- Always use 2 digits each number in date and time stamps in the listmode filename, example:
  *QC-Daily-2009.06.08.15.26.**02**_EM.164 (before: QC-Daily-2009.06.08.15.26.**2**_EM.164)*
- Add timestamp in millseconds to serial line log files.
- Add dead time constant to the scanner configuration file (scanit.ini) and write it in the listmode header., example:
  Dead time constant := 7.63E-6

Tag	Description
S14.1	Add milliseconds to the can start time in the listmode header for motion and listmode files synchronization
S14.2	Use 2 digits for date and time stamps in the listmode filename
S14.3	Add timestamp in millseconds to serial line log file

15. References

1. C. Comtat et al, “Image based resolution modeling for the HRRT OSEM reconstruction software”, IEEE MIC conference record, Dresden, 2008.

2. Merence Sibomana, Sune Høgild Keller, Søren Holm, Peter M. Bloomfield, Stephan Blinder, Søren Baarsgaard Hansen and Christian Michel, ” Component-Based Normalization for the HRRT for Sinogram-mode reconstruction”, IEEE MIC conference record, Orlando, 2009, to be published.