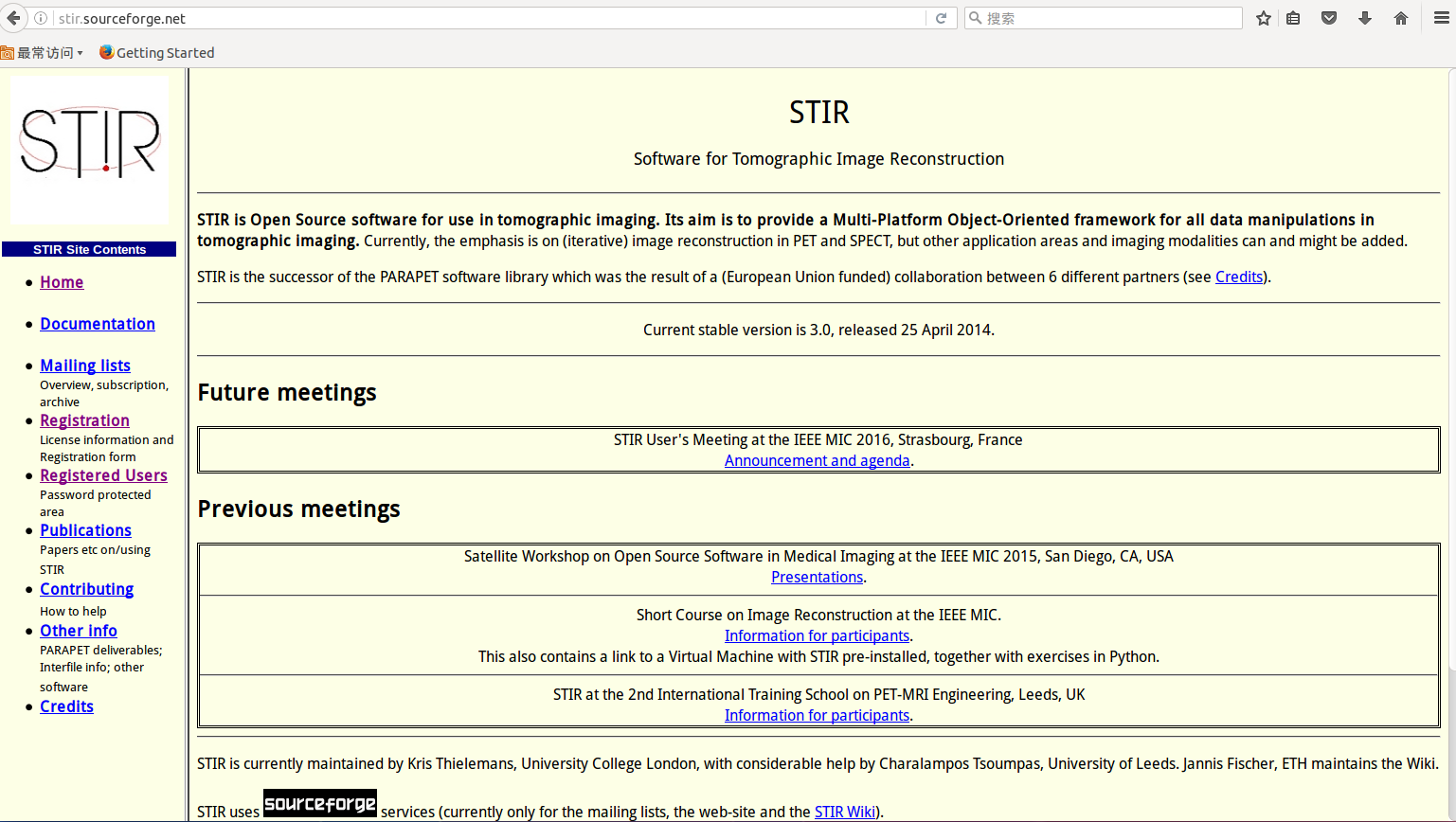
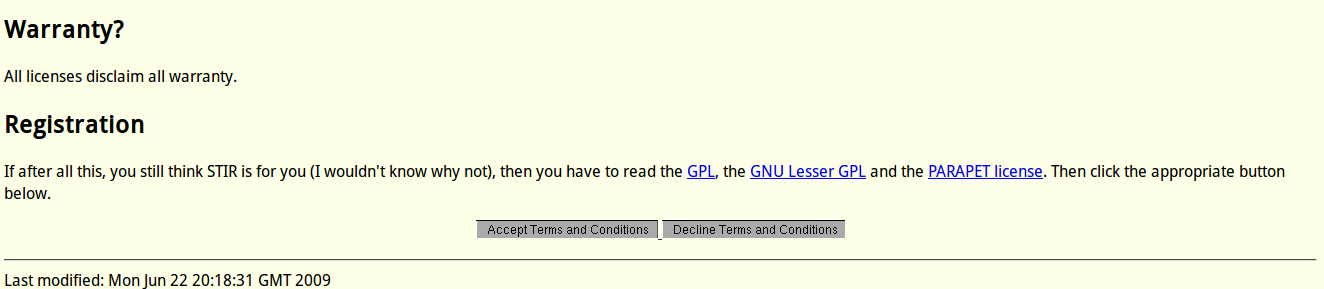
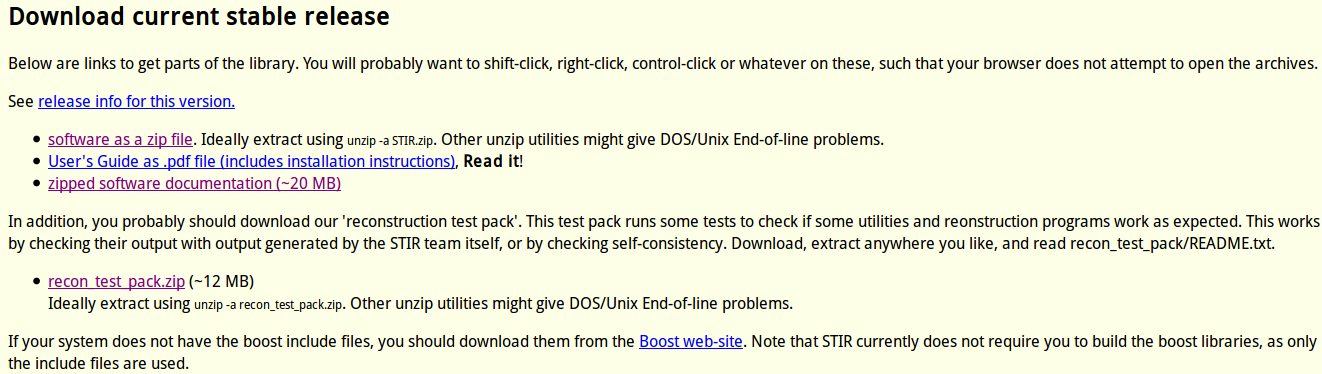
**STIR简介**

STIR安装步骤

第一步：注册。进入STIR网站，网址：<http://stir.sourceforge.net/>。点击页面左边的Registration，在跳转的页面最下方点击Accept Terms and Conditions,然后按照跳出页面的要求发送邮件进行注册。



第二步：下载安装包。在网页下点击Registered Users，输入用户名和密码。点击中间区域software as a zip file下载安装包，同时点击recon test pack.zip下载测试包。

除此之外，还需要BOOST安装包。可以在http://www.boost.org 下载。

ECAT7安装包。在http://www.opengatecollaboration.org/ECAT 下载

第三步：安装所有必要文件

创建目录

mkdir /home/twj/stir

把所有安装包放入该目录下

1）安装ECAT

mkdir ecat7

mv ecat.tar.gz ecat7 //将压缩包移动到ecat7目录下

cd ecat7

tar -zxf ecat.tar.gz //解压压缩包

cp Makefile.unix Makefile

make( 注 : 不能用 make -jN)

mkdir include

cp \*.h include

检查 libecat.a 在 lib 文件夹中 , 如果没有操作以下步骤

mkdir lib

cp libecat.a /lib

配置环境

export LD\_LIBRARY\_PATH=$LD\_LIBRARY\_PATH:/PATH\_TO/ecat7/lib

2) 解压BOOST

tar -zxf boost\_1\_61\_0.tar.bz2 //解压压缩包

准备工作完成后，安装STIR软件

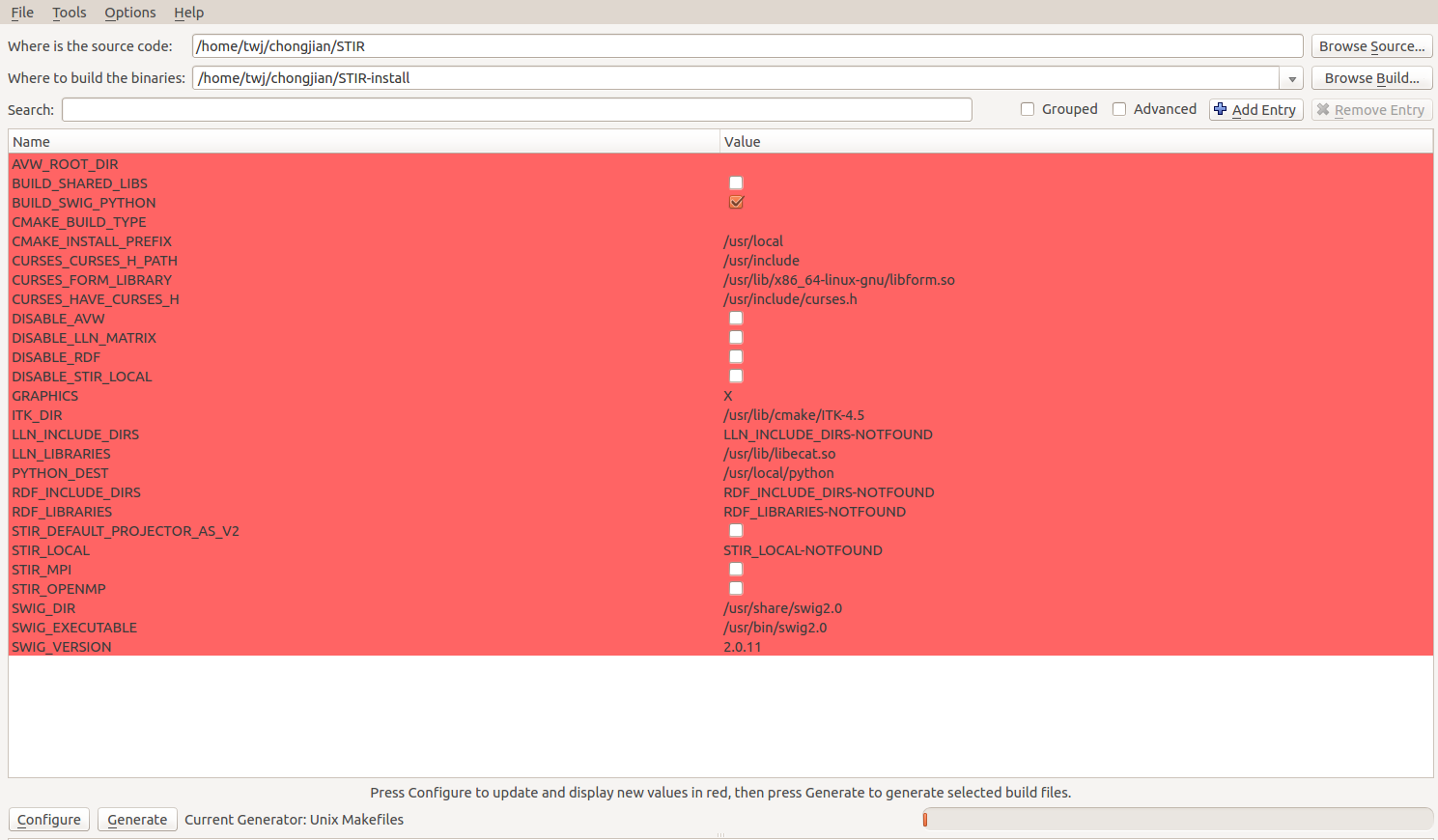
第四步：安装STIR

手动解压STIR.zip文件到当前目录下

cd ..

mkdir STIR-install //新建一个安装目录

打开cmake软件，如图

点击右上角的Browse Source，选择源文件（安装包）的路径

再点击Browser Build，选择安装目录

点击左下角Configure

完成后点击Generate

其中选项按照上图选择即可。

第五步：测试

解压测试包recon\_test\_pack.zip

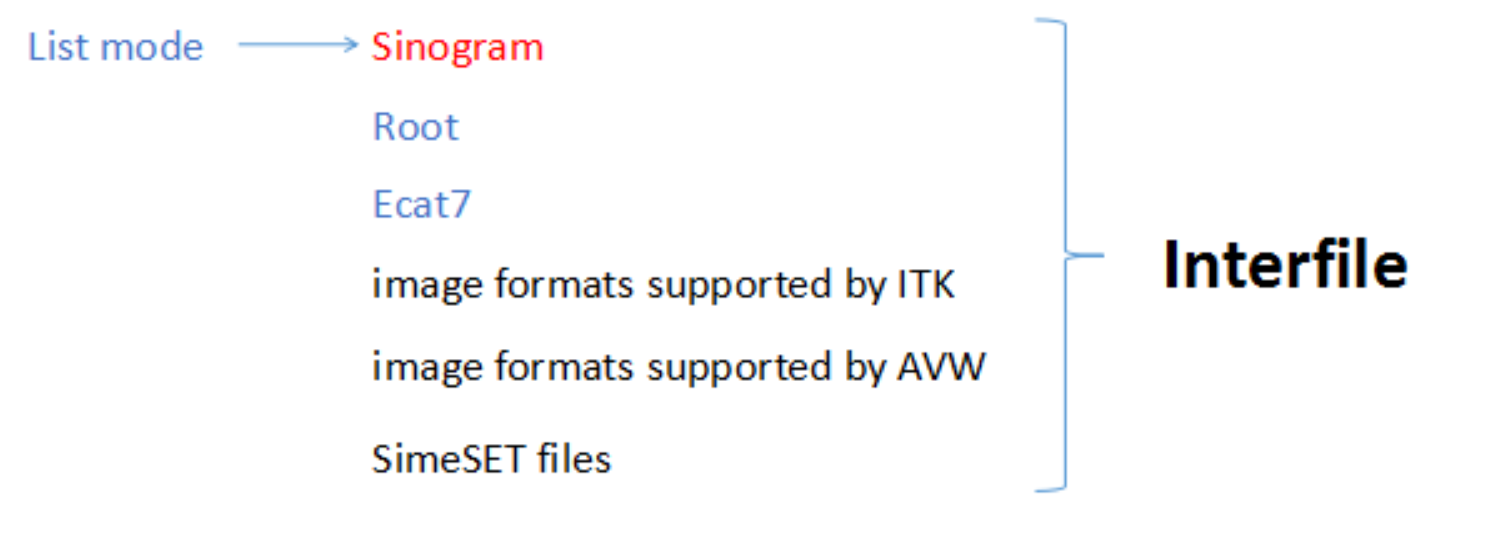
cd recon\_test\_pack

sh run\_tests.sh --nointbp ../STIR-install/bin

如果程序能够正常完成执行即安装成功，否则会报错。

使用步骤

**Part1:输入文件**

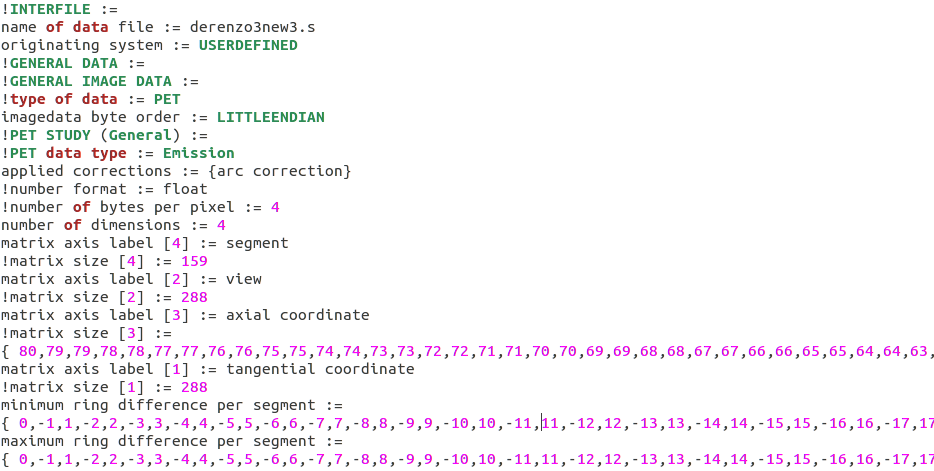


STIR在能够处理的特定格式的数据文件，该文件格式是interfile。其他格式文件也能读入，但依然存在各种各样的问题，在实际应用中可以先转换成interfile，再进行后续处理，如上图所示。其中list mode、sinogram、root、ecat7可以由GATE运行生成，因此在相对于其他几种格式的文件来说更为常用，而sinogram最为常用，这是由于输入的interfile与sinogram为同一数据。

What is Interfile?

Interfile=Header text file+data file（\*.hs+\*.s）

Header text file不仅给出对应的data file的数据格式、图像的像素数及层数，还给出了探测器相关结构参数，如下图所示：



其中，name of data file 为输入数据名称

imagedata byte order取决于PC（一般是little Endian）

number format为数据存储类型，取float

number of bytes per pixel取4

number of dimensions为4

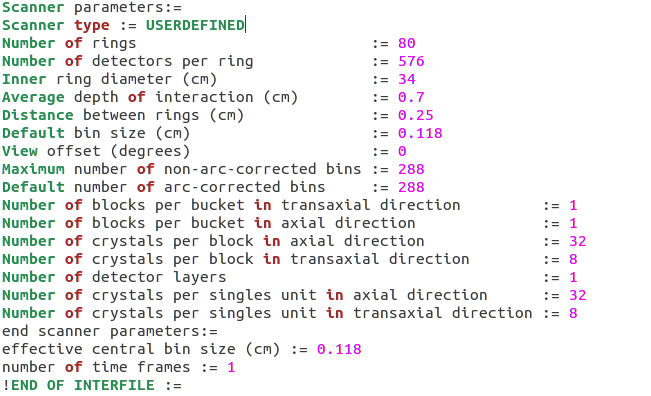
matrix size[1]和matrix size[2]是用来定义一条LOR的位置，分别定义它的距离和角度，其中角度对应实际角度中的0-180度。

matrix size[4]为数据分类，取环数2倍减1,如32环的数据则对应63个segment

matrix size[3]为轴向坐标，与matrix size[4]相关，表示每个segment的数据层数

minimum ring difference per segment和maximum ring difference per segment为两个数组，数组中的值一一对应，也与matrix segment[2]也必须一一对应，一般设置为{0，-1，1，······，-（最大环数差）+1，最大环数差-1，-（最大环数差），最大环数差}。

举个例子：如某一个5环的探测器，其oblique sinogram有25层，这25层按照环差可以分为9类，即为上述segment数量，分别对应环差为0,-1,1,-2,2,-3,3,-4,4，即为上述minmum ring difference和maximum ring difference，环差为0时有5层数据，环差为-1和1时有4层数据，环差为-2和2时有3层数据，环差为-3和3时有2层数据，环差为-4和4时有1层数据，即为上述axial coordinate。



该部分为环形探测器的结构参数，包括层数、每层晶体数量，环的内直径，环间距（轴向长度/层数）。除此以外，average depth of interaction取0.5-0.7都可以

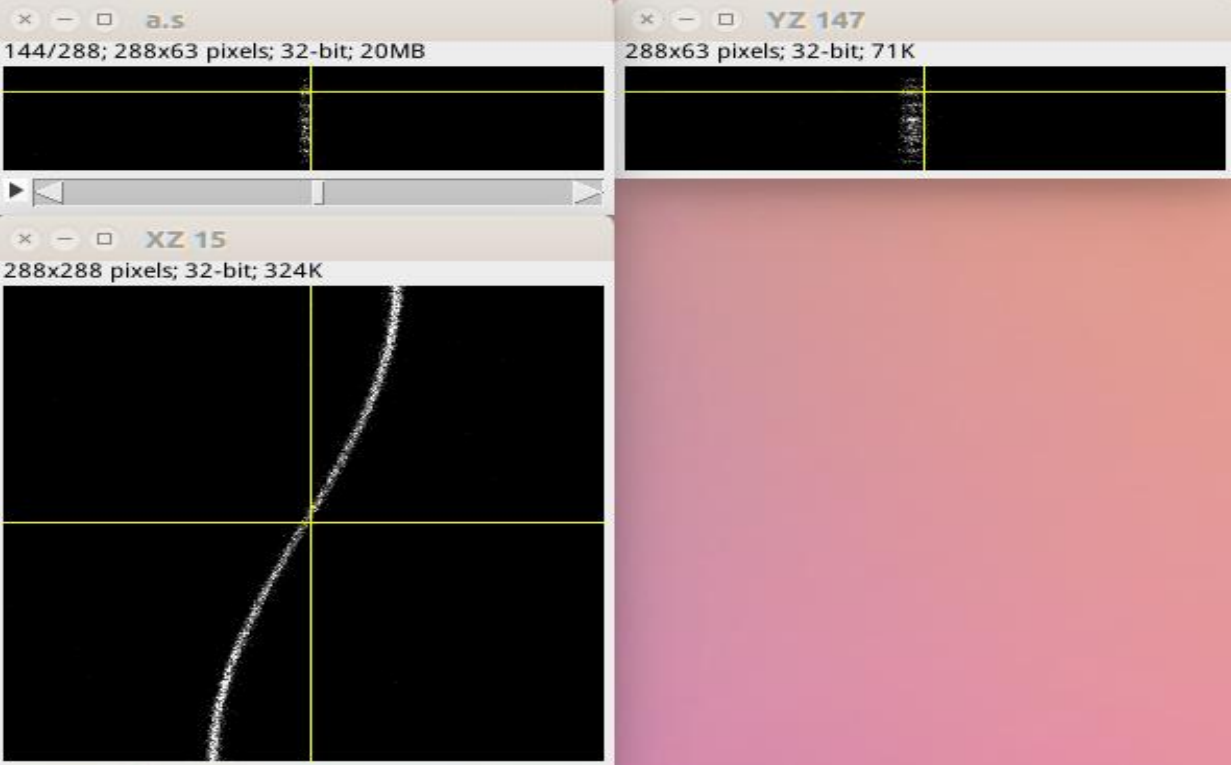
bin size和 effective central bin size取（内直径/bin个数）

view offset取0

maximum number of non-arc-corrected bins和default number of arc-corrected bins 取tangential coordinate

number of time frames取1。

data file文件按如图所示:



该文件数据内容与单层重排后的sinogram一致，其中左上图interfile内容，左下图即为sinogram数据。

**Part2:重建算法**

OSMAPOL

该程序执行IF-OSEM-OSL算法。类似于OSEM算法，但功能多于OSEM。它能够实现以下功能：

1. use of subsets 子集大小必须近似相等
2. inter-update filtering 原始图像更新之前进行滤波
3. inter-iteration filtering 迭代过程中（即完成某次图像更新后进行滤波）进行滤波
4. post-filtering 最后一次迭代后进行滤波
5. prior information
6. random order of the subsets in each iteration

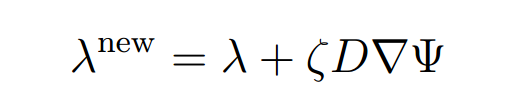
OSMAPOSLParameters :=  
;lines starting with semicolons are comments  
objective function type:= \  
PoissonLogLikelihoodWithLinearModelForMeanAndProjData  
PoissonLogLikelihoodWithLinearModelForMeanAndProjData Parameters:=  
; input, sensitivity and prior parameters here  
input file := projection\_data\_filename.hs %输入数据头文件名  
; use -1 to use the maximum available  
; maximum absolute segment number to process := 4 %迭代中使用的层数，一般定义为-1为使用全部层数  
; zero end planes of segment 0 := 1  
; keywords that specify the projectors to be used  
Projector pair type := Matrix  
Projector Pair Using Matrix Parameters :=  
; Use the PET Ray-tracing matrix.  
; This needs to be changed to SPECT UB when using SPECT data  
Matrix type := Ray Tracing  
Ray Tracing Matrix Parameters:=  
End Ray Tracing Matrix Parameters:=  
End Projector Pair Using Matrix Parameters := %投影数据相关定义  
; background (e.g. randoms)  
additive sinogram := 0 %是否有随机事件、散射事件等，若有可在此处添加相关头文件  
; sensitivity related keywords  
; time frame info used for dead-time calculation when using ECAT7  
;time frame definition filename:=  
;time frame number:= 1  
; normalisation and attenuation info  
; Bin Normalisation type:= None

recompute sensitivity := 1  
use subset sensitivities:=1 ; recommended  
; optional filename to store/read the sensitivity image  
; (if use subset sensitivity is off)  
; sensitivity filename:=  
; optional filename to store/read the subsensitivities  
; use %d where you want the subset-number (a la printf)  
subset sensitivity filenames:= sens\_%d.hv %对不同子集采用不同的灵敏度定义，相应定义从文件中读取  
; keywords for specifying the prior information  
prior type := None %提供先验信息  
; next keywords can be used to specify image size, but will be removed  
; they are ignored when using an initial image  
zoom := 1  
; use --1 for default sizes that cover the whole field of view  
XY output image size (in pixels) := -1  
end PoissonLogLikelihoodWithLinearModelForMeanAndProjData Parameters:=  
; set output file format, if omitted a default value will be used  
Output file format := Interfile  
Interfile Output File Format Parameters :=  
; byte order := little-endian  
; number format := signed integer  
; number of bytes per pixel := 2  
End Interfile Output File Format Parameters :=  
initial estimate:= initial\_image\_filename.hv  
enforce initial positivity condition:=1   
number of subsets:= 6  
start at subset:= 0  
number of subiterations:= 30  
start at subiteration number:=1  
output filename prefix := out\_file  
save estimates at subiteration intervals:= 2  
uniformly randomise subset order:= 1 %输出文件的格式及参数，定义了子集及迭代次数  
; keywords that specify the filtering that occurs after every subiteration  
; warning: do not normally use together with a prior  
inter-iteration filter subiteration interval := 4  
inter-iteration filter type := Separable Cartesian Metz  
; keywords below will depend on the filter type (see text)  
separable cartesian metz filter parameters :=  
x-dir filter fwhm (in mm) := 6  
y-dir filter fwhm (in mm) := 6  
z-dir filter fwhm (in mm) := 6  
; use some sharpening here as example (not really recommended though)  
x-dir filter metz power := 2  
y-dir filter metz power := 2  
z-dir filter metz power := 2  
end separable cartesian metz filter parameters :=  
; keywords that specify the filtering that occurs at the end  
; of the reconstruction  
post-filter type := None  
; keywords that specify the filtering that occurs before  
; multiplying with the update image  
inter-update filter subiteration interval := 4  
; would have to be filled in.  
inter-update filter type := None  
map model := additive %定义是否使用滤波器以及所使用滤波器的参数（取决于滤波器的类型）  
; keywords for preventing too drastic (multiplicative) updates  
; below just set to their default values  
maximum relative change := 3.40282e+38  
minimum relative change := 0 %乘法更新阈值  
; enabling this will write the multiplicative update images  
; every sub-iteration  
write update image := 0  
END :=

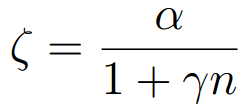
以上各参数中加红部分为最基本参数设置，必须定义完整。其余参数可根据实际情况加入。

OSSPS

该程序执行有序子集的梯度下降算法。



其中λ为待评估值，Ψ为目标函数，D为预处理器（取决于Ψ），ζ为松弛参数。

由可对ζ进行设置，n为迭代次数除以子集数。  
OSSPSParameters :=

; sample file for OSSPS

; parameters used here are for illustrative purposes only

; i.e. they are not recommended values

; Note:

; some variables here are indicated with ${VAR}

; These can be initialised as environment variables before

; running POSSPS. E.g. in bash you could do

; MAXSEG=2 NUMSUBS=6 other\_variables\_here POSSPS POSSPS.par

; Alternatively, copy this file and edit it obviously.

objective function type:=PoissonLogLikelihoodWithLinearKineticModelAndDynamicProjectionData

PoissonLogLikelihoodWithLinearKineticModelAndDynamicProjectionData Parameters:=

input file := fwd\_dyn\_from\_p0005-p5.S

; if next disabled, defaults to maximum segment number in the file

maximum absolute segment number to process := ${MAXSEG}

projector pair type := Matrix

Projector Pair Using Matrix Parameters :=

Matrix type := Ray Tracing

Ray tracing matrix parameters :=

End Ray tracing matrix parameters :=

End Projector Pair Using Matrix Parameters :=

; if the next parameter is disabled,

; the sensitivity will be computed using the normalisation object

sensitivity filename:= sens\_POSSPS.img

; if next is set to 1, sensitivity will be recomputed

; and written to file (if "sensitivity filename" is set)

recompute sensitivity := 1

; background term (i.e. randoms and scatter)

additive sinograms := 0

; see STIR doc: use 1 if segment 0 has only ring difference 0

; zero\_segment 0\_end\_planes:= 0

zero end planes of segment 0:= 0

;prior type:= Quadratic

;quadratic prior parameters :=

;penalisation factor := .5

;only 2D:=1

;kappa filename := kappas\_a\_la\_fessler.hv

;end quadratic prior parameters:=

; patlak related files

Kinetic Model type := Patlak Plot

Patlak Plot Parameters :=

time frame definition filename := time.fdef

starting frame := 23

calibration factor := 9432.31

blood data filename := plasma.if

Time Shift := 0

In total counts := 1

;In correct scale := 0

end Patlak Plot Parameters :=

end PoissonLogLikelihoodWithLinearKineticModelAndDynamicProjectionData Parameters:=

output filename prefix:=POSSPS

; iteration scheme

; Number of subsets should be a divisor of num\_views/4

number of subsets:=${NUMSUBS}

; Use for starting the numbering from something else than 1

start at subiteration number:=1

; Use if you want to start from another subset than 0 (but why?)

start at subset:= 0

number of subiterations:= ${ITER}

save estimates at subiteration intervals:= ${SAVITER}

;write update estimate := 0

; if next is disabled, defaults to image full of 1s (but that's not good for OSSPS)

; in particular, make sure it has the correct scale

initial estimate := indirect\_Patlak.img

enforce initial positivity condition := 1

; additional regularisation

;inter-iteration filter subiteration interval:= 0

;inter-iteration filter type := none

; here start OSSPS specific values

; values to use for the 'precomputed denominator'

; if you do not specify the following keyword, the 'precomputed denominator'

; will be computed automatically (and saved)

; use the following if you have it already (e.g. from previous run)

; note: setting the value to 1 will use an images full of ones (which is not a good idea!)

; precomputed denominator := my\_precomputed\_denominator.hv

; specify relaxation scheme

; lambda = relaxation\_parameter/ (1+relaxation\_gamma\*(subiteration\_num/num\_subsets)

relaxation parameter := 1

relaxation gamma:=.1 %借助上述公式对ζ进行设置

END :=

FBP2D

该程序执行single slice rebinning和FBP算法。

fbp2dparameters :=  
input file := input.hs  
output filename prefix := output  
; output image parameters  
; zoom defaults to 1  
zoom := 1  
; image size defaults to whole FOV  
xy output image size (in pixels) := 180  
; can be used to call SSRB first  
; default means:  
; if no axial compression, use 3  
; otherwise, use 1  
;num segments to combine with ssrb := -1 %可以设置是否使用SSRB

; filter parameters, default to pure ramp  
alpha parameter for ramp filter := 1  
cut-off for ramp filter (in cycles) := 0.5 %滤波器使用海明窗，此处根据公式对α和f进行设置

; keywords that specify the filtering that occurs at the end  
; of the reconstruction  
post-filter type := None  
end :=

该过程中使用到的SSRB算法的设置，最完整的参数包括输入输出文件名、结合层数数量（必须是奇数）或者结合的视图数量、标准化（可选）、最大待处理层数（默认为总层数）

FBP3DRP

与FBP2D基本类似，默认使用的滤波器是“pure”Colsher filter。

**Part3:输出文件**

Herder text file+data file(\*.hv+\*.v)

Header文件中包含了对应的数据文件名、数据格式、图像像素及层数等信息。

!INTERFILE :=

name of data file := true1r.v

!GENERAL DATA :=

!GENERAL IMAGE DATA :=

!type of data := PET

imagedata byte order := LITTLEENDIAN

!PET STUDY (General) :=

!PET data type := Image

process status := Reconstructed

!number format := float

!number of bytes per pixel := 4

number of dimensions := 3

matrix axis label [1] := x

!matrix size [1] := 91

scaling factor (mm/pixel) [1] := 8.8

matrix axis label [2] := y

!matrix size [2] := 91

scaling factor (mm/pixel) [2] := 8.8

matrix axis label [3] := z

!matrix size [3] := 63

scaling factor (mm/pixel) [3] := 2.2

first pixel offset (mm) [1] := -396

first pixel offset (mm) [2] := -396

first pixel offset (mm) [3] := 0

number of time frames := 1

!END OF INTERFILE :=

**Part4:三维重建应用实例**

本示例中完成从GATE仿真结果的root和sinogram数据到interfile数据的转换，并分别利用STIR中的FBP3DRP和OSEM完成重建。

分别利用FBP和OSEM算法对图像进行重建，ECAT系统探测器及phantom参数如下：

探测内半径41.2cm，32环，轴向长度15.52cm，72个模块，每个模块64个晶体，晶体大小是30mm×4.4mm×4.75mm，phantom为derenzo模型，该模型只有一层，厚度为6mm，每层512×512个像素，每个像素0.6mm×0.6mm，因此整个phantom直径30cm左右，其中最小点7.2mm。

**头文件中参数设置如下：**

!INTERFILE :=

name of data file := ceshi72.s

originating system := USERDEFINED

!GENERAL DATA :=

!GENERAL IMAGE DATA :=

!type of data := PET

imagedata byte order := LITTLEENDIAN

!PET STUDY (General) :=

!PET data type := Emission

applied corrections := {arc correction}

!number format := float

!number of bytes per pixel := 4

number of dimensions := 4

matrix axis label [4] := segment

!matrix size [4] := 63

matrix axis label [2] := view

!matrix size [2] := 288

matrix axis label [3] := axial coordinate

!matrix size [3] := { 32,31,31,30,30,29,29,28,28,27,27,26,26,25,25,24,24,23,23,22,22,21,21,20,20,19,19,18,18,17,17,16,16,15,15,14,14,13,13,12,12,11,11,10,10,9,9,8,8,7,7,6,6,5,5,4,4,3,3,2,2,1,1 }

matrix axis label [1] := tangential coordinate

!matrix size [1] := 288

minimum ring difference per segment := { 0,-1,1,-2,2,-3,3,-4,4,-5,5,-6,6,-7,7,-8,8,-9,9,-10,10,-11,11,-12,12,-13,13,-14,14,-15,15,-16,16,-17,17,-18,18,-19,19,-20,20,-21,21,-22,22,-23,23,-24,24,-25,25,-26,26,-27,27,-28,28,-29,29,-30,30,-31,31 }

maximum ring difference per segment := { 0,-1,1,-2,2,-3,3,-4,4,-5,5,-6,6,-7,7,-8,8,-9,9,-10,10,-11,11,-12,12,-13,13,-14,14,-15,15,-16,16,-17,17,-18,18,-19,19,-20,20,-21,21,-22,22,-23,23,-24,24,-25,25,-26,26,-27,27,-28,28,-29,29,-30,30,-31,31 }

Scanner parameters:=

Scanner type := USERDEFINED

Number of rings := 32

Number of detectors per ring := 576

Inner ring diameter (cm) := 82.4

Average depth of interaction (cm) := 0.7

Distance between rings (cm) := 0.485

Default bin size (cm) := 0.286

View offset (degrees) := 0

Maximum number of non-arc-corrected bins := 288

Default number of arc-corrected bins := 288

Number of blocks per bucket in transaxial direction := 1

Number of blocks per bucket in axial direction := 1

Number of crystals per block in axial direction := 32

Number of crystals per block in transaxial direction := 8

Number of detector layers := 1

Number of crystals per singles unit in axial direction := 32

Number of crystals per singles unit in transaxial direction := 8

end scanner parameters:=

effective central bin size (cm) := 0.286

number of time frames := 1

!END OF INTERFILE :=

**OSEM参数设置如下：**

OSMAPOSLParameters :=

objective function type:= PoissonLogLikelihoodWithLinearModelForMeanAndProjData

PoissonLogLikelihoodWithLinearModelForMeanAndProjData Parameters:=

input file := your3d.hs

;zero end planes of segment 0:= 0 ; segment 0 is has direct and indirect planes

; if disabled, defaults to maximum segment number in the file

;maximum absolute segment number to process := -1

projector pair type := Matrix

Projector Pair Using Matrix Parameters :=

Matrix type := Ray Tracing

Ray tracing matrix parameters :=

number of rays in tangential direction to trace for each bin := 10

do symmetry 90degrees min phi := 1

do symmetry 180degrees min phi := 1

End Ray tracing matrix parameters :=

End Projector Pair Using Matrix Parameters :=

; Bin Normalisation type := from projdata

; Bin Normalisation From ProjData :=

; normalisation projdata filename:= ${MULTFACTORS}

; End Bin Normalisation From ProjData:=

;additive sinogram := ${ADDSINO}

xy output image size (in pixels) := 180

zoom := 1

use subset sensitivities:=1

;subset sensitivity filenames:= my\_DSTE\_sens\_2D\_PM\_s14\_%d.hv

recompute\_sensitivity:=1

prior type := quadratic

Quadratic Prior Parameters:=

penalisation factor := 0

;weights:={{{0,1,0},{1,0,1},{0,1,0}}}

END Quadratic Prior Parameters:=

end PoissonLogLikelihoodWithLinearModelForMeanAndProjData Parameters:=

output filename prefix := derenzo3new3

number of subsets:= 8

start at subset:= 0

number of subiterations:= 8

save estimates at subiteration intervals:=8

;report objective function values interval := 1

END :=

**FBP3D参数设置如下：**

fbp3drpparameters :=

input file := your3d.hs

output filename prefix := ceshi72

; all following parameters are optional

;;;;;;; output image parameters

zoom := 1

; defaults to cover whole FOV

xy output image size (in pixels) := 180

;maximum absolute segment number to process := 63

;;;;;;; parameters for initial image

; you can use an existing image (but be careful about the scale!)

; image to be used for reprojection := some\_image

; or you can use FBP on 2D data. following are reconstruction parameters for 2D

; default means: call SSRB only if no axial compression is already present

num segments to combine with ssrb := 1

; filter parameters, default to pure ramp

alpha parameter for ramp filter := 1

cut-off for ramp filter (in cycles) := 0.5

;;;;;;;; parameters for Colsher filter

alpha parameter for colsher filter in axial direction := 1

cut-off for colsher filter in axial direction (in cycles) := 0.5

alpha parameter for colsher filter in planar direction := 1

cut-off for colsher filter in planar direction (in cycles) := 0.5

; define colsher on finer grid. The higher the number, the better (but slower)

stretch factor for colsher filter definition in axial direction := 2

stretch factor for colsher filter definition in planar direction := 2

; allow less padding. DO NOT USE

transaxial extension for fft := 1

axial extension for fft := 1

;;;;;;;;; other parameters

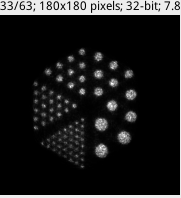
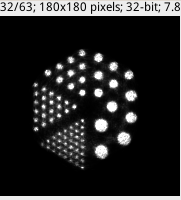
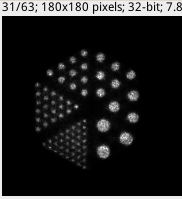
save intermediate images := 0

display level := 0

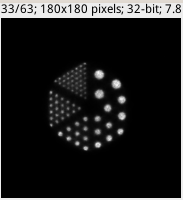
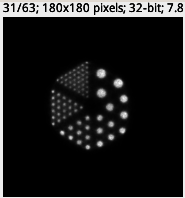
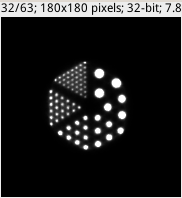
end :=

**重建结果如下：**

sinogram数据的迭代重建结果（空间中的三层）

****

root数据的迭代重建结果



FBP3DRP重建结果

