MODIFICATION OF CT QUALITY ASSURANCE PHANTOM FOR PET/CT ALIGNMENT AND PET RESOLUTION

A Thesis

Submitted to the Graduate Faculty of the
Louisiana State University and
Agricultural and Mechanical College
in partial fulfillment of the
requirements for the degree of
Master of Science

in

The Department of Physics and Astronomy

by
Prashanth K Nookala
B.Tech., Jawaharlal Nehru Technological University, 2001

May 2005

ACKNOWLEDGEMENTS

It is a pleasure to thank the many people who made this thesis possible. I thank my advisor, Dr. Kenneth L Matthews II, for spending his valuable time to teach and guide me, without whom this work would not have been possible. I sincerely thank him for his suggestions and patience in motivating and nurturing me thr

I thank my parents, brothers and other family members back in India, for being there in my hard times and for their encouragement, constant support, love and blessings without which I would not be here.

I also thank Department of Physics and Astronomy for supporting me financially by granting assistantships through out the course of study.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Radiotherapy treatment planning utilizing PET and CT is rapidly gaining acceptance in oncology. A limiting factor of the dual modality is the PET/CT alignment. A small error in PET/CT alignment may result in giving large doses of radiation to healthy tissues as a result of poor treatment planning. For this purpose, regular quality assurance testing of PET/CT must be performed. Separate QA procedures and phantoms have been developed for the two different modalities. In particular, many existing phantoms cannot be used for both modalities, which is a requirement for evaluating PET/CT alignment. Our goal is to evaluate several existing phantom designs to evaluate their utility for checking PET/CT alignment. The three phantoms investigated are a Gammex 464 phantom, a Triple-Line Source PET phantom, and a Hot Sphere PET phantom. The PET phantoms are unmodified the Gammex 464 phantom is modified to perform PET/CT alignment. The Gammex 464 phantom is typically used for routine quality assurance of CT scanners. Several CT parameters are determined with this phantom before and after modification. Then PET/CT alignment testing is performed using this modified CT phantom and the two other phantoms. Three methods have been used for analyzing the PET/CT images to measure the PET/CT alignment errors. The methods are the Manual method which calculates the alignment error from hand-drawn profiles, the Maximum-Pixel Value method which measures the error based on the pixel value of the objects in the PET/CT images, and the Curve-fitting method, which measures the alignment error by getting the best fit values for the object profiles. The Curve-fitting method also estimates the PET resolution from apparent size of objects in the phantoms.

Our PET/CT alignment data and results suggest that the Maximum-Pixel Value method for the modified phantom with acrylic insert is a good choice for measuring the PET/CT alignment error, providing a reasonable balance between computational analysis effort and measurement precision.

CHAPTER 1

INTRODUCTION

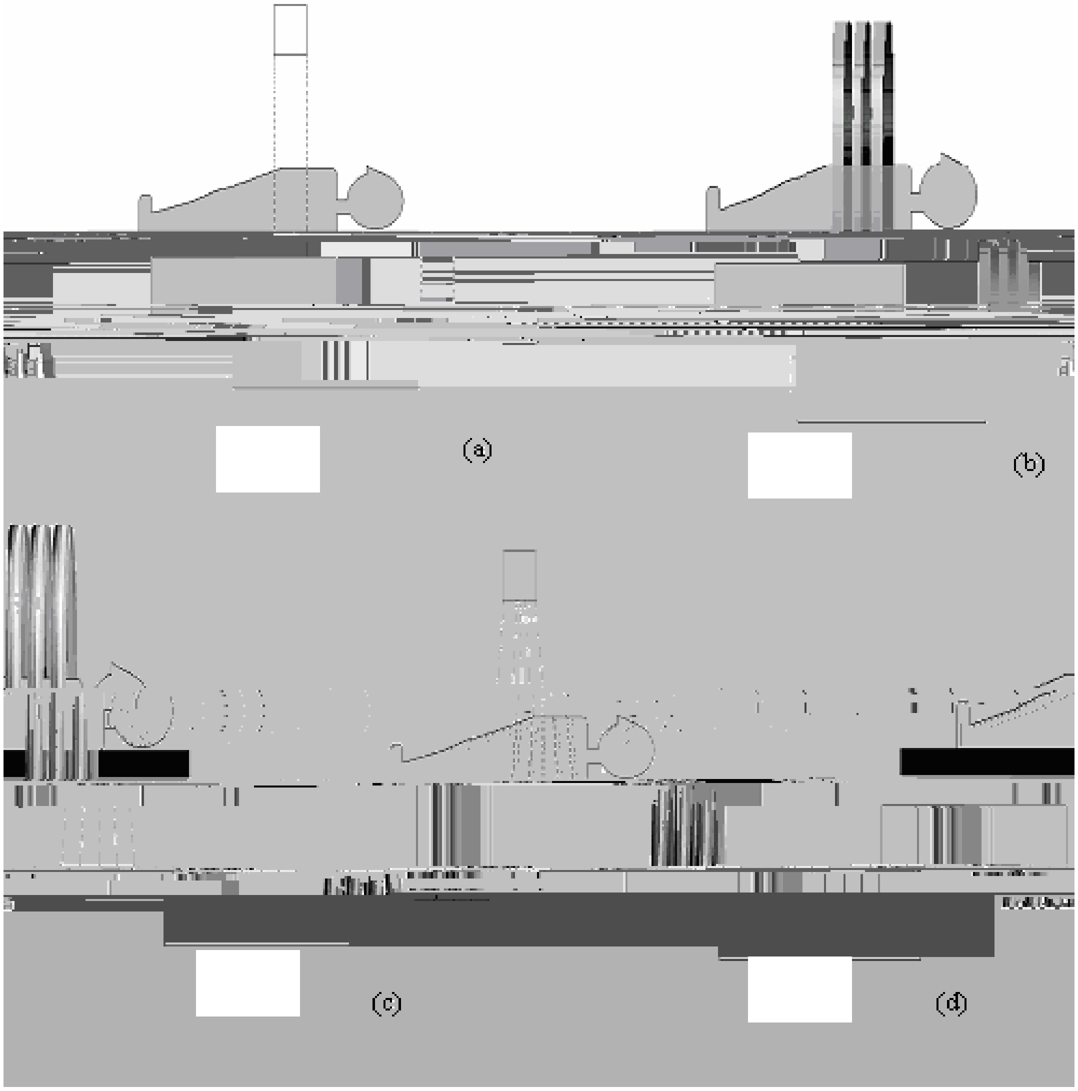
1.1 Objectives

CHAPTER 2

BACKGROUND PHYSICS

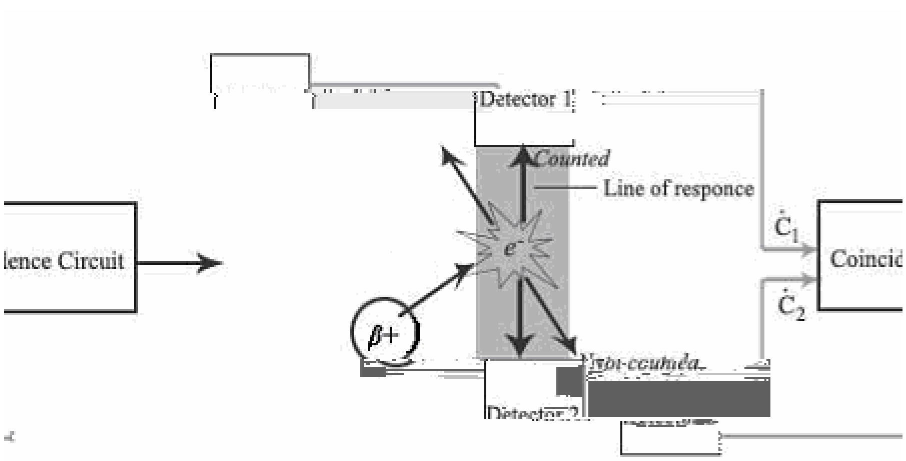
2.1 Computed Tomography (CT)

$$\frac{\mu_t - \mu_w}{\mu_w}$$



2.2 Positron Emission Tomography (PET)

Ű


Detector 1
Counted
Line of responce
e-
β+
Detector 2
lence Circuit
Coincid
Ċ1
Ċ2

2.3 PET/CT Dual Modality

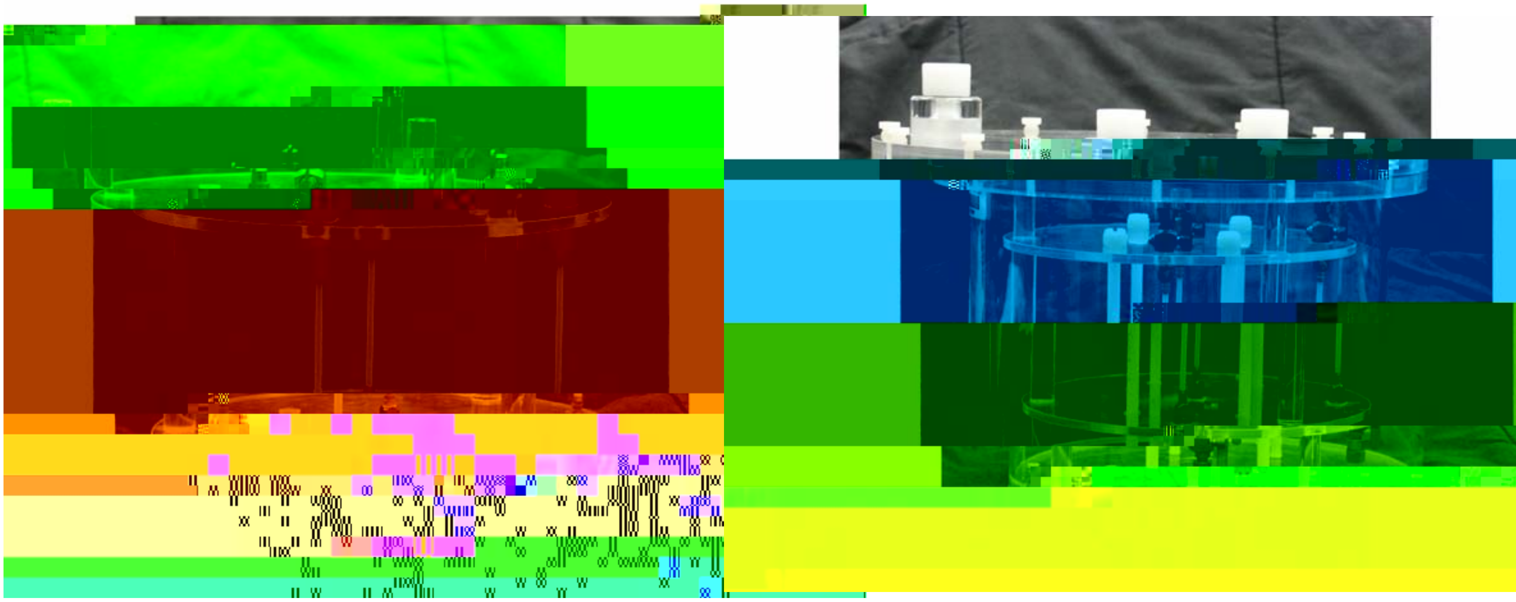
CHAPTER 3

MATERIALS

3.1 ACR CT Accreditation Phantom, Gammex 464

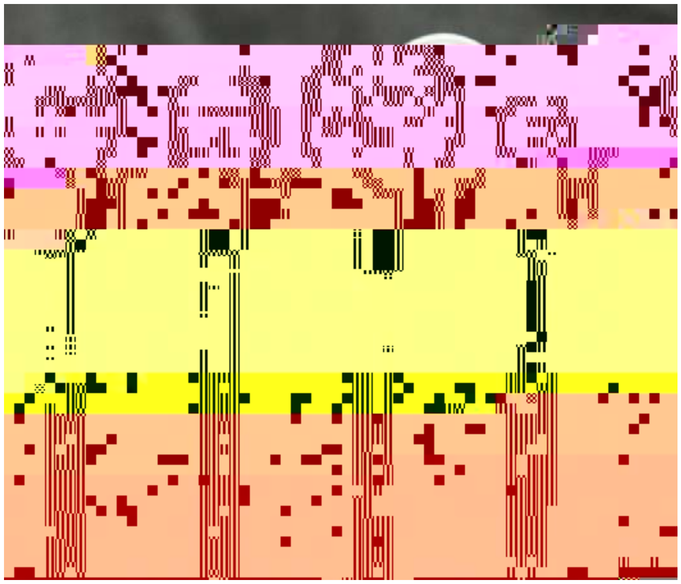
Parameter	Value

3.2 Modified ACR CT Accreditation Phantom, Gammex 464



Parameter	Value

3.5 Hot Sphere Phantom



CHAPTER 4

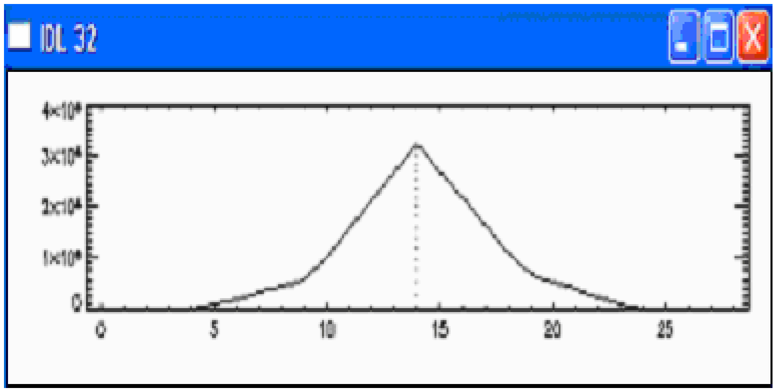
METHODS

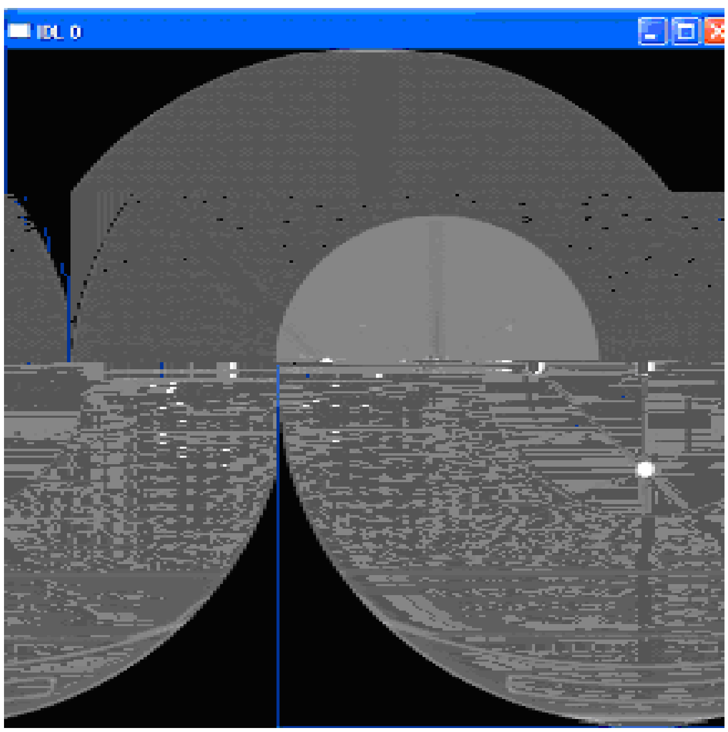
4.1 CT Quality Assurance

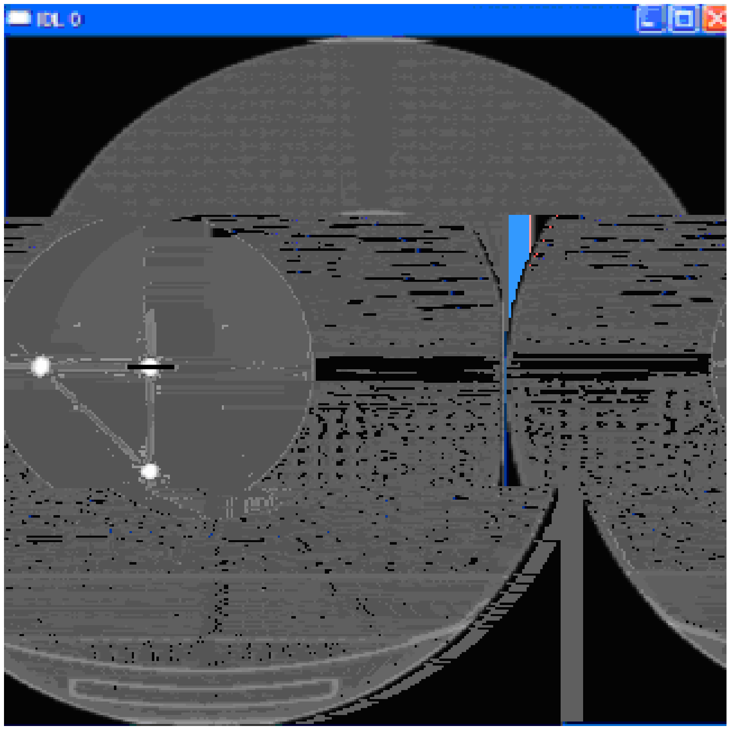
4.2 PET/CT Alignment Testing

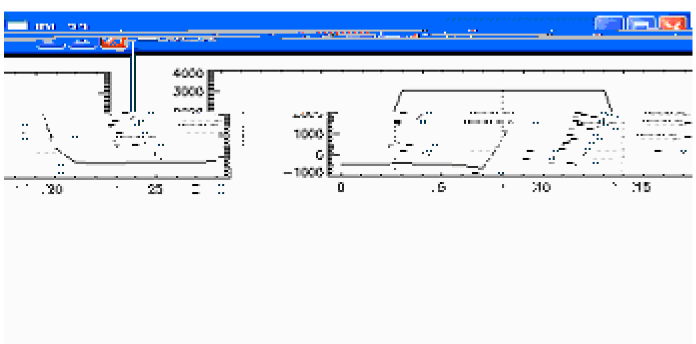
4.2.2 PET/CT Image Analysis

4.2.2.1 Manual Method

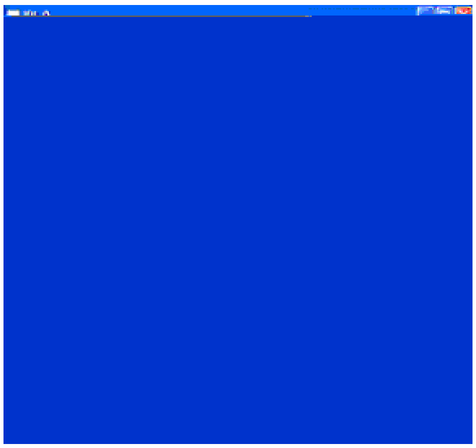

IDL 32
4×10⁴
3×10⁴
2×10⁴
1×10⁴
0
0
5
10
15
20
25


IDL 0


IDL 0

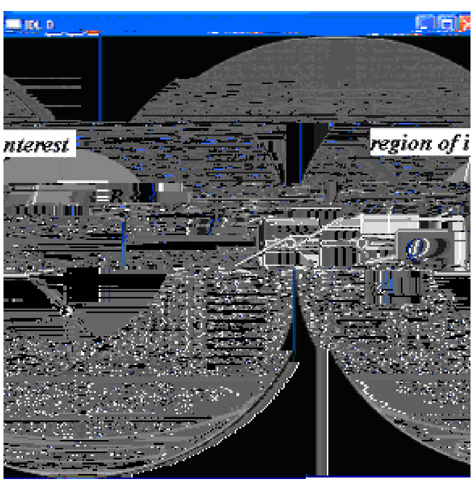

4000
3000
1000
0
−1000
20
25
0
5
10
15

4.2.2.2 Maximum-Pixel Value Method

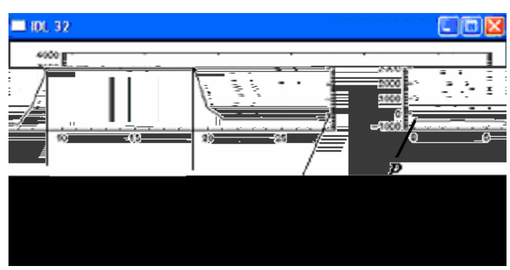


(a)

(b)



(a)



(b)

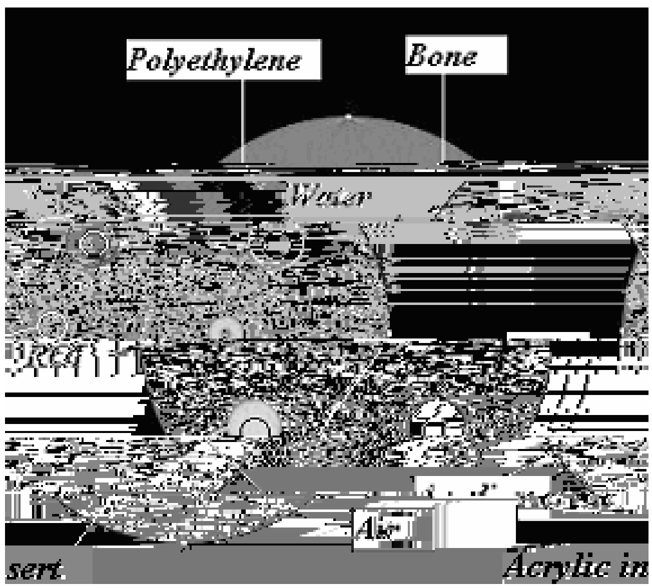
4.3 Determination of PET Resolution

σ

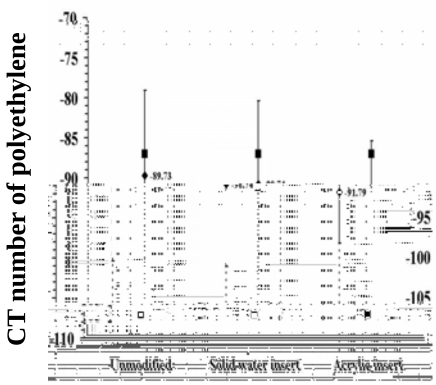
σ

$$\sqrt{FWHM \quad - \; S_o}$$

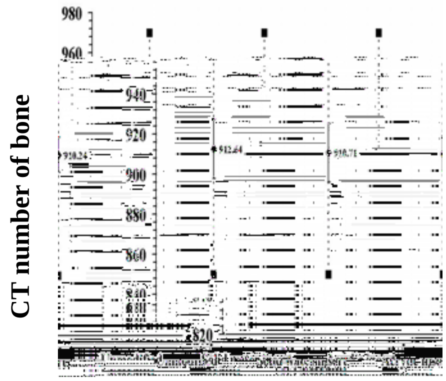
CT Acquisition Parameter	Value



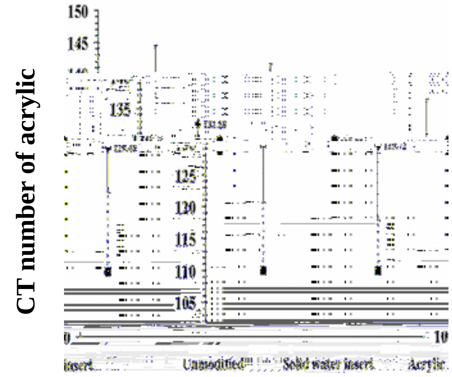
Material	Minimum CT Number (HU)	Maximum CT Number (HU)



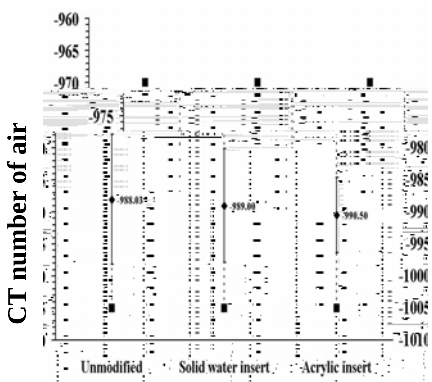
(a)



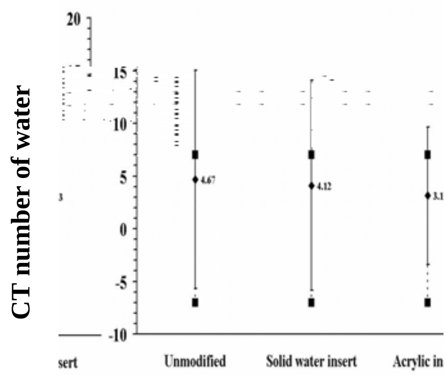
(b)



(c)



(d)



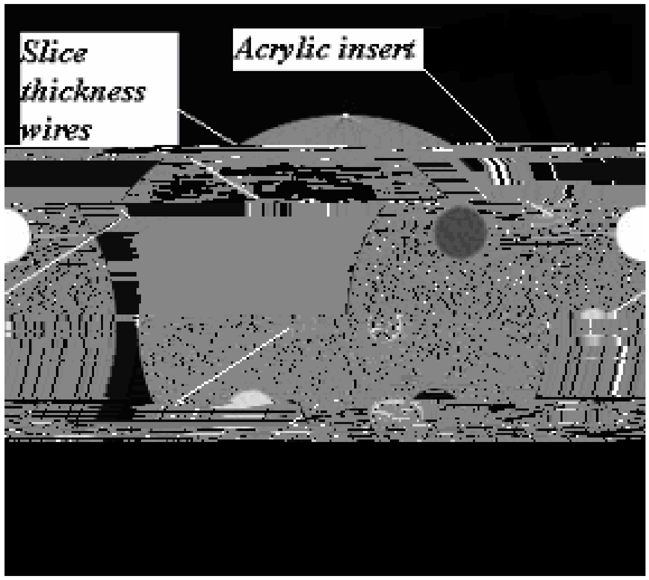
(e)

5.1.2 Slice Thickness

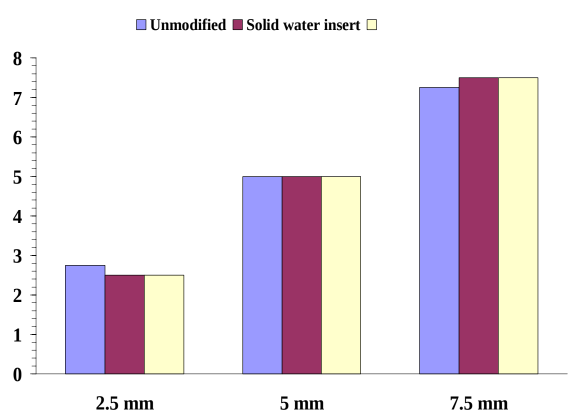
5.1.2.1 Determination of Slice Thickness at 120 kVp

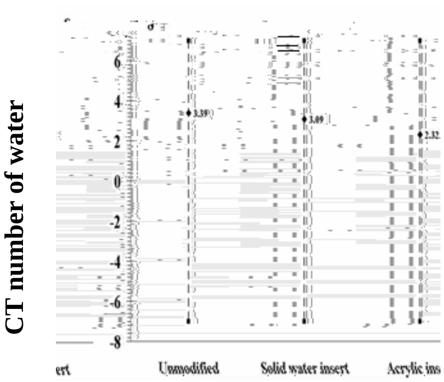
5.1.2.2 Verification of CT Number vs. kVp

5.1.3 Low Contrast Resolution

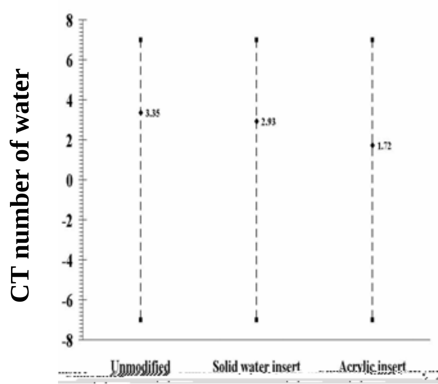


CT Acquisition Parameter	Value

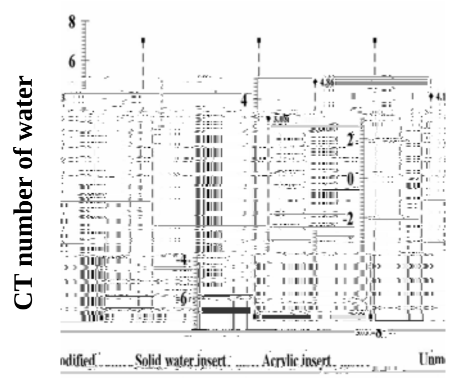




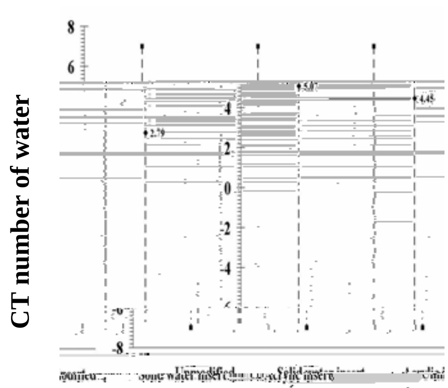
(a)



(b)

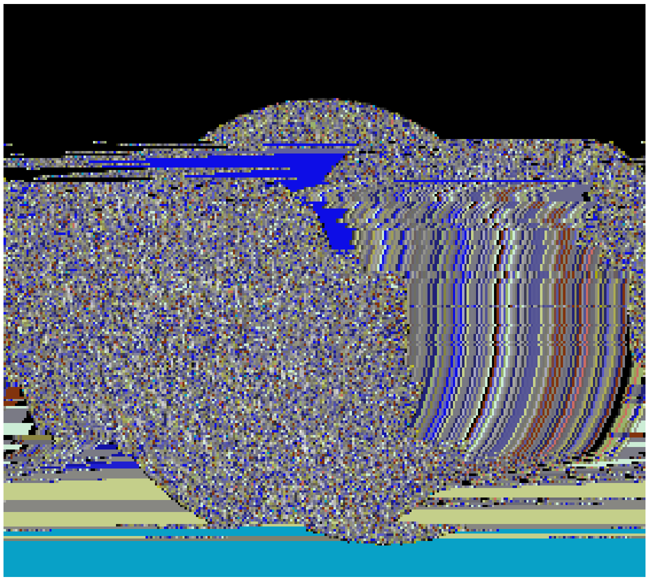


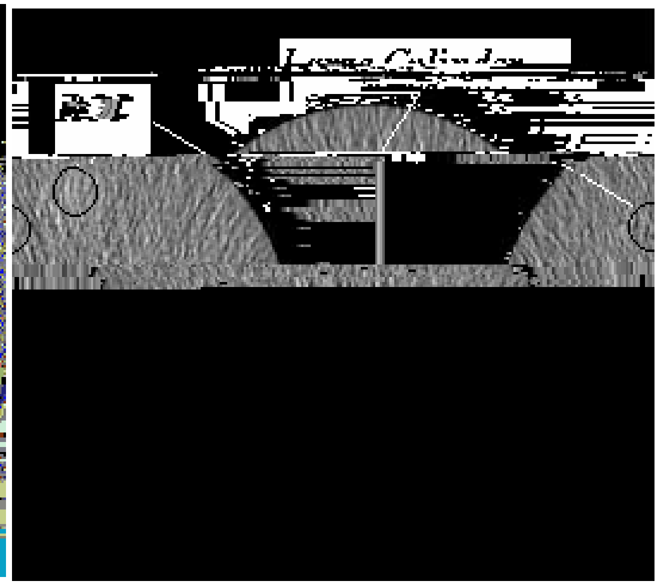
(c)



(d)

CT Acquisition parameter	Value

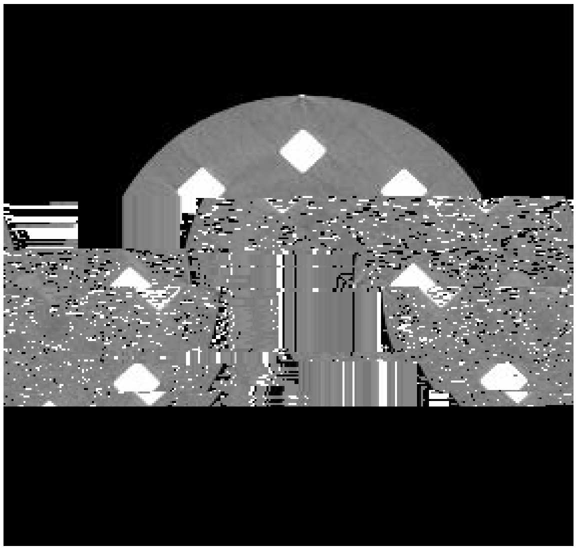
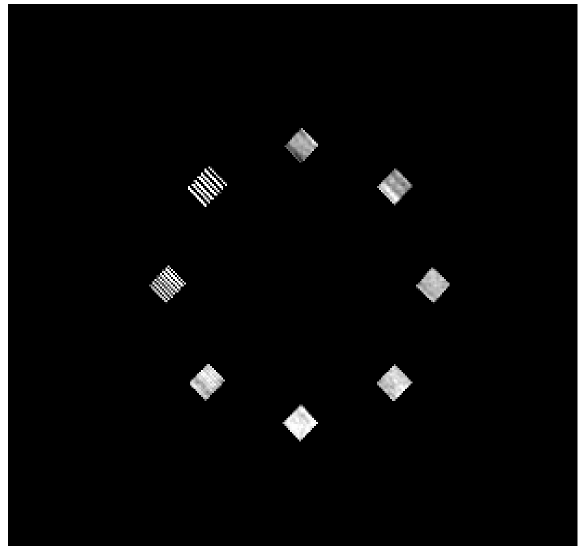




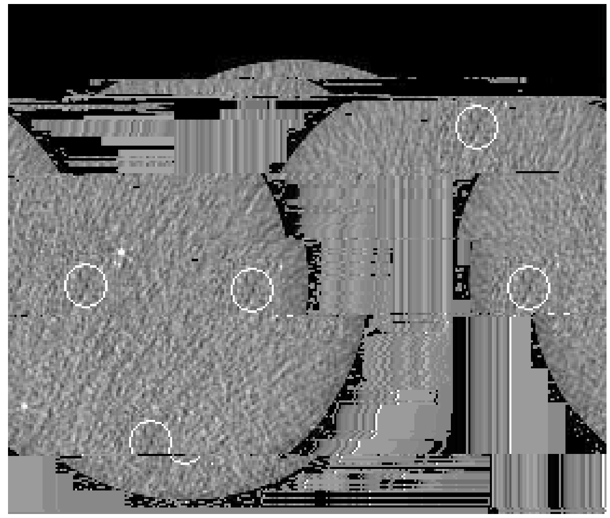
Phantom	Diameters of visible cylinders (mm)	CT number (large cylinder) (HU)	CT number (next largest cylinder) (HU)

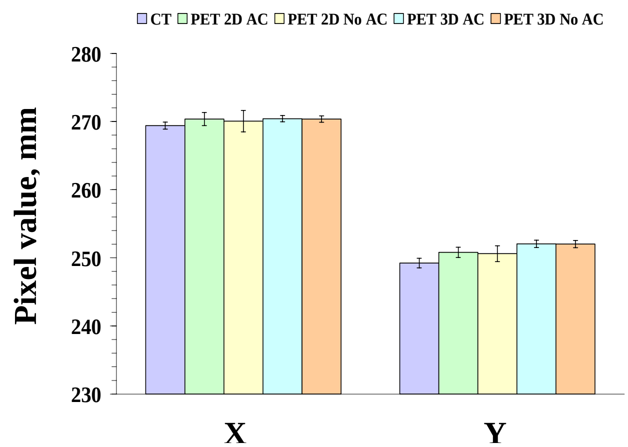
5.1.4 High Contrast Resolution

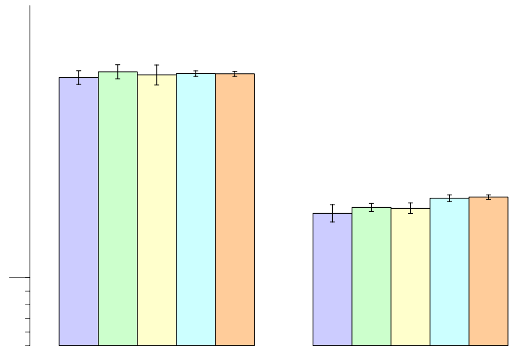
5.1.5 Uniformity

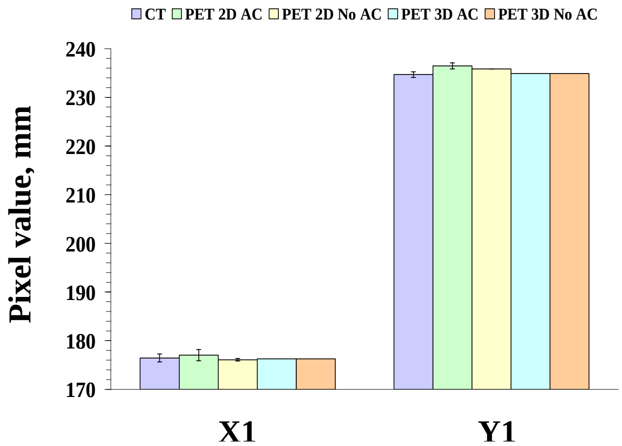



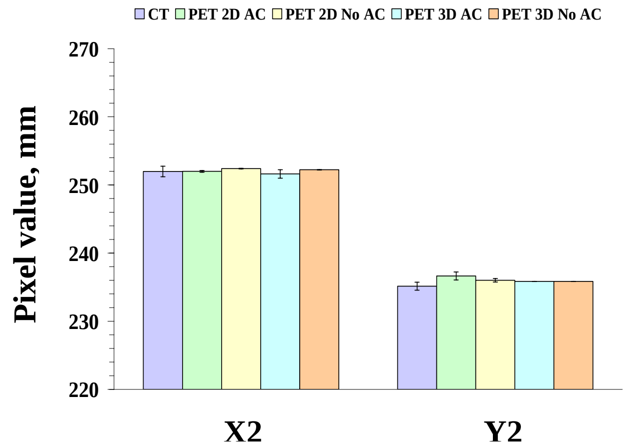

6'o clock position
9'o clock position
Uniformity



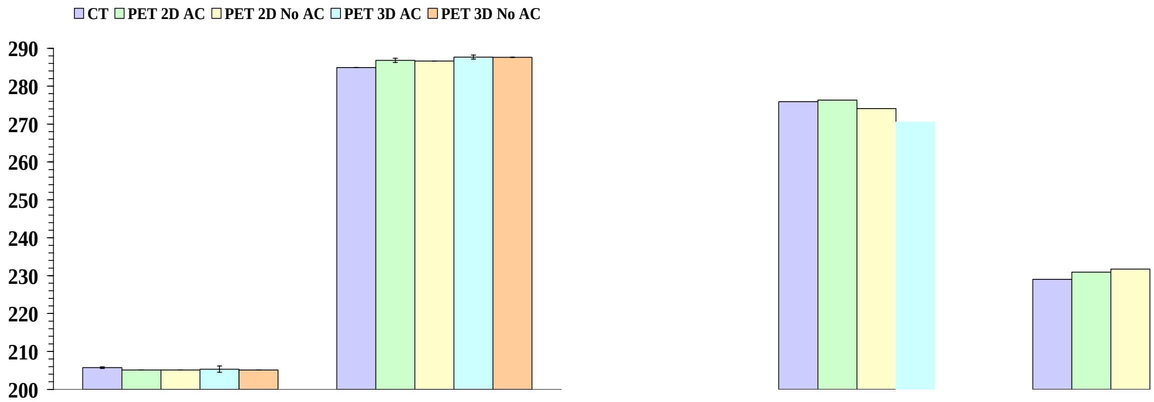

CT
PET 2D AC
PET 2D No AC
PET 3D AC
PET 3D No AC
Pixel value, mm
280
270
260
250
240
230
X
Y

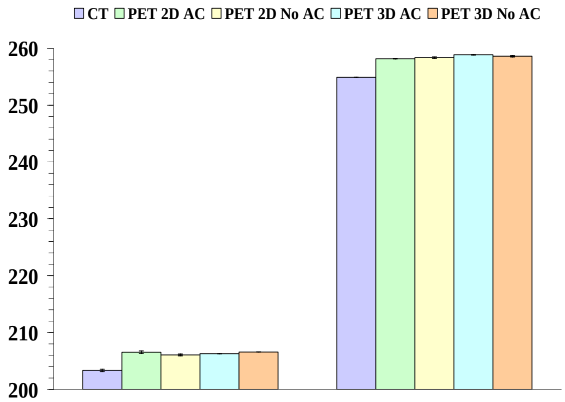


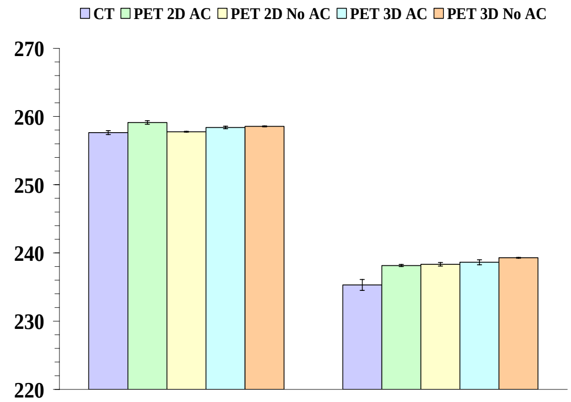

CT
PET 2D AC
PET 2D No AC
PET 3D AC
PET 3D No AC
Pixel value, mm
240
230
220
210
200
190
180
170
X1
Y1
(a)

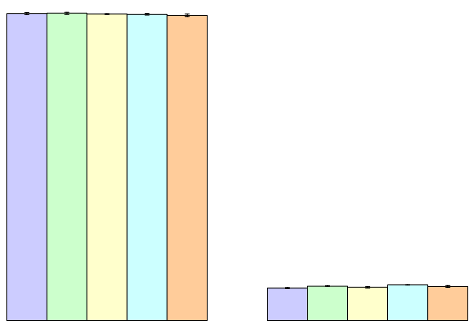

CT
PET 2D AC
PET 2D No AC
PET 3D AC
PET 3D No AC
Pixel value, mm
270
260
250
240
230
220
X2
Y2
(b)

5.2.1.4 Triple-line Source

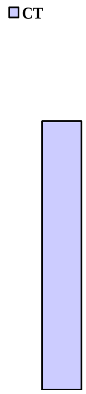

CT
PET 2D AC
PET 2D No AC
PET 3D AC
PET 3D No AC
290
280
270
260
250
240
230
220
210
200


CT
PET 2D AC
PET 2D No AC
PET 3D AC
PET 3D No AC
260
250
240
230
220
210
200


CT
PET 2D AC
PET 2D No AC
PET 3D AC
PET 3D No AC
270
260
250
240
230
220

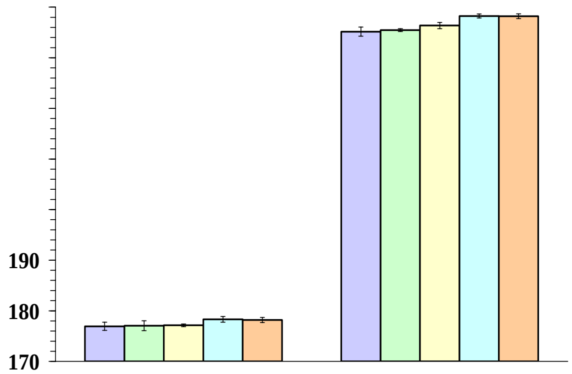


5.2.2.2 Modified Gammex Phantom with Solid Water Insert



Alignment error (mm)	2D		3D	
	Attenuation Correction	No Attenuation Correction	Attenuation Correction	No Attenuation Correction
Δ _ σ	_	_	_	_
Δ _ σ	_	_	_	_

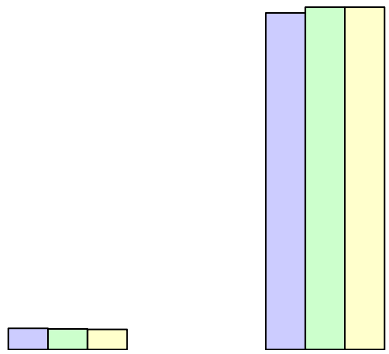
5.2.2.3 Triple-line Source Phantom



190
180
170

5.2.2.4 Triple-Line Source Phantom Rotated by 45°

°

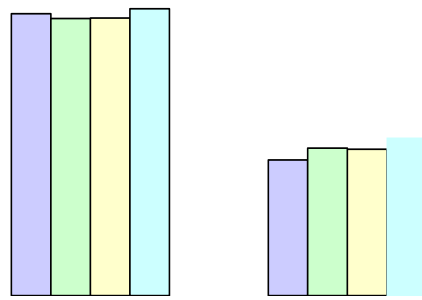


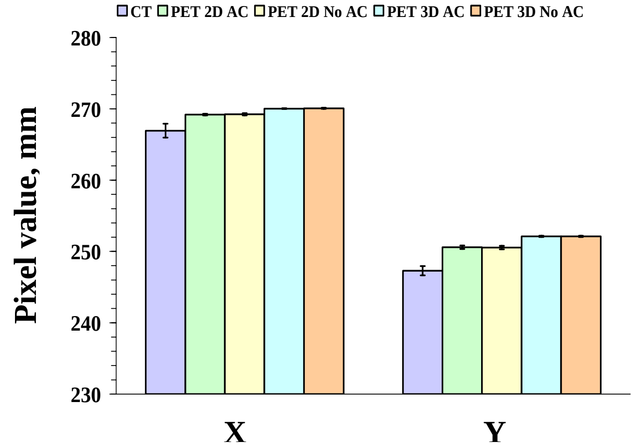
[redacted]

Diameter of sphere (mm)	Alignment error (mm)	2D		3D	
		Attenuation Correction	No Attenuation Correction	Attenuation Correction	No Attenuation Correction
	Δ _ σ	_	_	_	_
	Δ _ σ	_	_	_	_
	Δ _ σ	_	_	_	_
	Δ _ σ	_	_	_	_
	Δ _ σ	_	_	_	_
	Δ _ σ	_	_	_	_
	Δ _ σ	_	_	_	_
	Δ _ σ	_	_	_	_
	$\Delta \overline{X}$ _ σ	_	_	_	_
	$\Delta \overline{Y}$ _ σ	_	_	_	_

5.2.3 Results from Curve-fitting Method

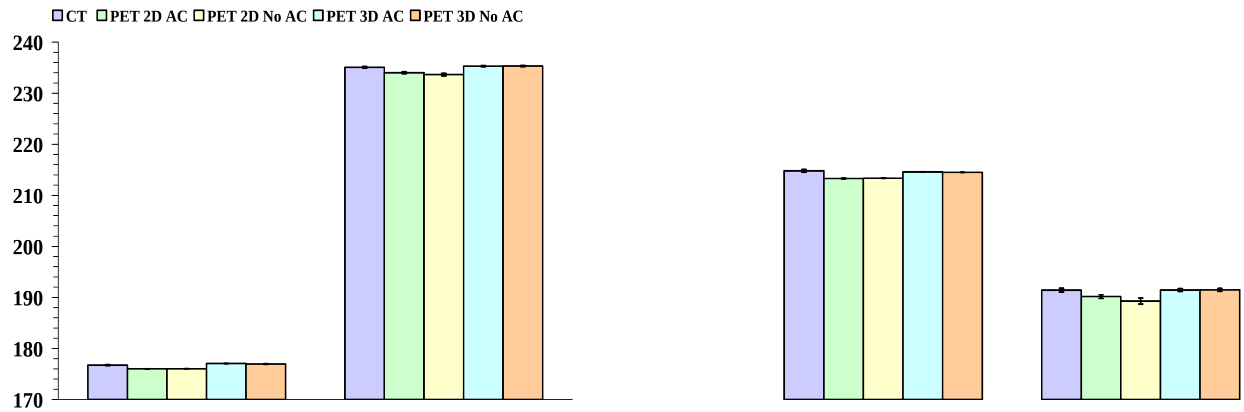
5.2.3.1 Modified Gammex Phantom with Acrylic Insert



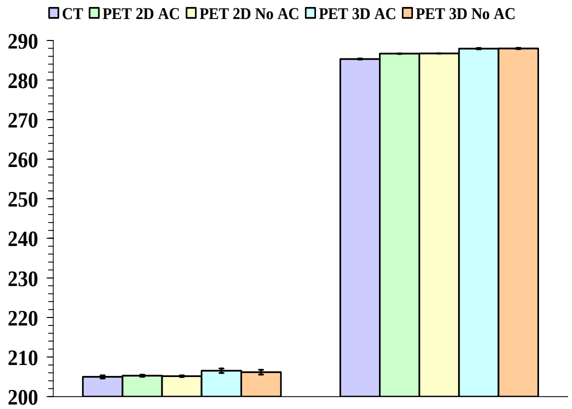


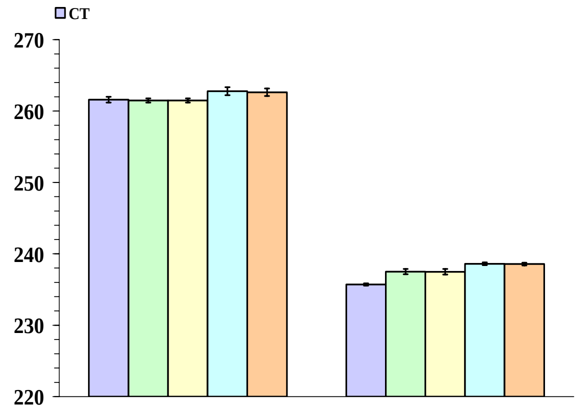
Alignment error (mm)	2D		3D	
	Attenuation Correction	No Attenuation Correction	Attenuation Correction	No Attenuation Correction
Δ _ σ	_	_	_	_
Δ _ σ	_	_	_	_

5.2.3.3 Triple-Line Source Phantom


CT
PET 2D AC
PET 2D No AC
PET 3D AC
PET 3D No AC
240
230
220
210
200
190
180
170

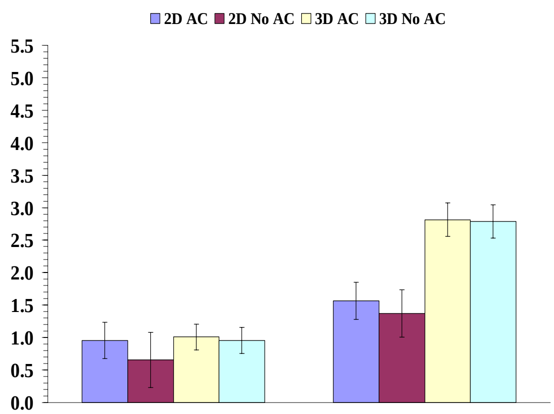
5.2.3.4 Triple-Line Source

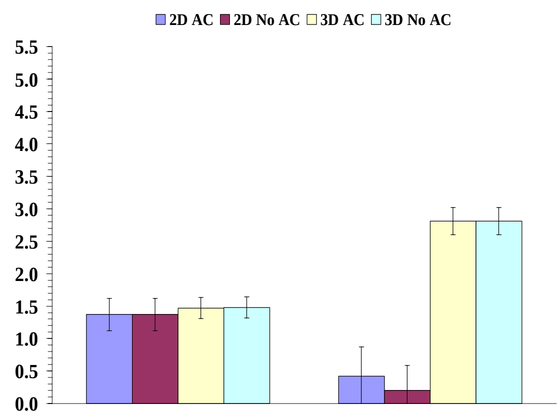

CT
PET 2D AC
PET 2D No AC
PET 3D AC
PET 3D No AC
290
280
270
260
250
240
230
220
210
200

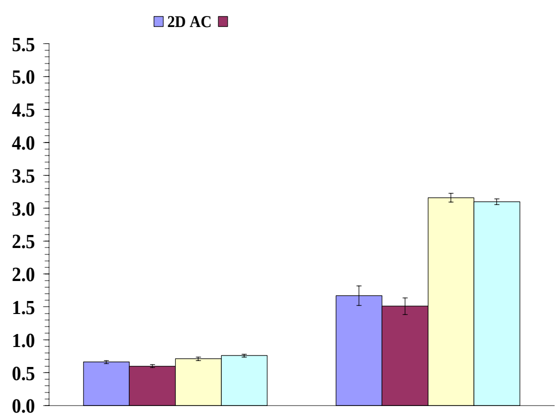

CT
270
260
250
240
230
220

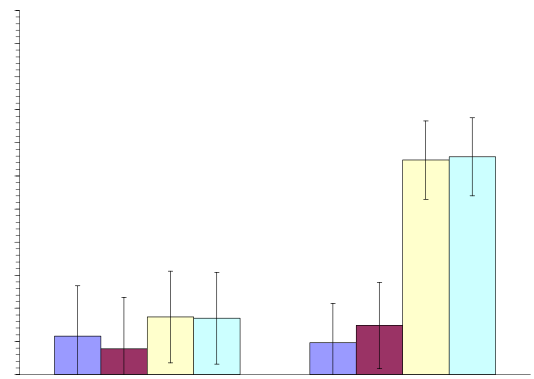
Diameter of sphere (mm)	Alignment error (mm)	2D		3D	
		Attenuation Correction	No Attenuation Correction	Attenuation Correction	No Attenuation Correction
	Δ _ σ	_	_	_	_
	Δ _ σ	_	_	_	_
	Δ _ σ	_	_	_	_
	Δ _ σ	_	_	_	_
	Δ _ σ	_	_	_	_
	Δ _ σ	_	_	_	_
	Δ _ σ	_	_	_	_
	Δ _ σ	_	_	_	_

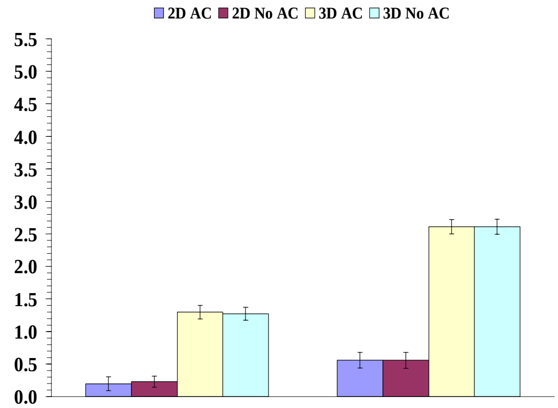
Δ —


2D AC
2D No AC
3D AC
3D No AC
5.5
5.0
4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0


2D AC
2D No AC
3D AC
3D No AC
5.5
5.0
4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0

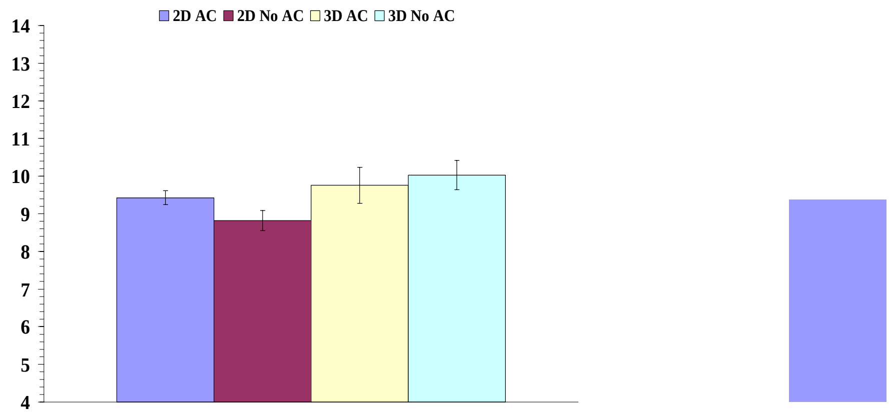

2D AC
5.5
5.0
4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0




2D AC
2D No AC
3D AC
3D No AC
5.5
5.0
4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0

$$\sqrt{\quad mm \quad - \quad mm \quad}$$

Phantom	Object size (diameter)	Average measured object size (FWHM)



CHAPTER 6

SUMMARY AND CONCLUSIONS

CT Parameter tested	Unmodified Gammex Phantom	Modified Gammex Phantom with acrylic Insert	Modified Gammex Phantom with solid water insert

REFERENCES

APPENDIX A

IDL PROGRAMS FOR MANUAL METHOD

```
pro ctt1
!p.multi[2] = 4
!p.charsize=1.5
!x.style = 3
window, 0, xsize=512, ysize=512
```

APPENDIX B

IDL PROGRAMS FOR MAXIMUM-PIXEL VALUE METHOD

```
pro TRIAL_ct_tlsp_rotn

!p.multi[2] = 4
!p.charsize=1.5
!x.style = 3

OPENW,1,'C:\Documents and Settings\prashanth\Desktop\point.txt'
read,PROMPT='Enter X LOW  :',X_LOW
read,PROMPT='Enter X HIGH :',X_HIGH
read,PROMPT='Enter Y LOW  :',Y_LOW
read,PROMPT='Enter Y HIGH :',Y_HIGH
read,PROMPT='Enter X:',X
read,PROMPT='Enter Y:',Y
x=fix(x)
y=fix(y)
for i=X,Y do begin
    IMAGE='F:\ISGARS\0\'+string(i)
    words = STRSPLIT(IMAGE, ' ', /EXTRACT)
    CT_IMAGEPATH=words[0]+words[1]

printf,1,CT_IMAGEPATH,'(',CT_X(CT_IMAGEPATH,X_LOW,X_HIGH,Y_LOW,Y_HIGH),'
,',CT_Y(CT_IMAGEPATH,X_LOW,X_HIGH,Y_LOW,Y_HIGH),')
endfor
close,1

!p.multi = 0
end
```

```
FUNCTION CT_X,IMAGEPATH,R,M,S,N
A=read_dicom(IMAGEPATH)
X=0.90*MAX(A)
for i=M,R-1,-1.0 do begin
    for j=N,S-1,-1.0 do begin
        if A[i,j] GT X then begin
        k=i
        i=10.0
        j=10.0
        ENDIF
    endfor
endfor
```

```
pro TRIAL_pet
```

```
        k=i
        i=10.0
        j=10.0
        ENDIF
    endfor
endfor

for i=R,M-1.0 do begin
    for j=S,N-1.0 do begin
        if C[i,j] GT X then begin
        l=i
        i=1000.0
        j=1000.0
        ENDIF
    endfor
endfor

x_PET_256=(l+(k-l)/2)
RETURN,x_PET_256
END
```

```
FUNCTION PET_Y,IMAGEPATH,R,M,S,N
A=read_dicom(IMAGEPATH)
magnifiedImg = CONGRID(A, 614, 614, /INTERP)
C=intarr(564,566)
C=magnifiedImg[49:613,47:613]
X=0.60*MAX(C)
```

APPENDIX C

IDL PROGRAMS FOR CURVE-FITTING METHOD

```
PRO tlsp_ct_fit
!p.multi[2] = 4
!p.charsize=1.5
!x.style = 3

OPENW,2,'C:\Documents and Settings\prashanth\Desktop\point1.txt'
read,PROMPT='Enter X LOW  :',X_LOW
read,PROMPT='Enter X HIGH :',X_HIGH
read,PROMPT='Enter Y LOW  :',Y_LOW
read,PROMPT='Enter Y HIGH :',Y_HIGH
read,PROMPT='Enter X:',X
read,PROMPT='Enter Y:',Y
x=fix(x)
y=fix(y)

for i=X,Y do begin
    IMAGE='F:\ISGARS\0\'+string(i)
    words = STRSPLIT(IMAGE, ' ', /EXTRACT)
    CT_IMAGEPATH=words[0]+words[1]
printf,2,CT_IMAGEPATH,'(',tlsp_ct_x(CT_IMAGEPATH,x_low,x_high,y_low,y_hi
gh),tlsp_ct_y(CT_IMAGEPATH,x_low,x_high,y_low,y_high),')'
endfor

close,2
!p.multi = 0
end
```

```
function tlsp_ct_x,ct_imagepath,x_low,x_high,y_low,y_high
A=read_dicom(ct_imagepath)

mask12=mask_tlsp_new(x_low)
m=0.0
n=0.0
difference=fltarr(12,12)

for m=x_low,x_high do begin
    for n=y_low,y_high do begin
```

```
for i=0,11 do begin
  for j=0,11 do begin
   difference[i,j]=mask12[i,j]-A[m+i,n+j]
  endfor
endfor
 avg=average1(difference)
 if(avg lt 5.0)and(avg gt -5.0)crence[i,j]=mask12[i,j]-A[m+i,n+j]
```

```
a=read_dicom('F:\ISGARS\0\450')
mask=fltarr(12,12)
mask[0:11,0:11]=a[174:185,236:247]
return,mask
END
```

```
function average1,difference
sum=0.0
average=0.0

for i=0,11.0 do begin
    for j=0,11.0 do begin
        sum=sum+difference[i,j]
    endfor
endfor

average=sum/144.0
return,average
end
```

```
pro FIT_resltn
!p.multi[2] = 4
!p.charsize=1.5
!x.style = 3

OPENW,1,'C:\Documents and Settings\prashanth\Desktop\point.txt'
read,PROMPT='Enter X LOW  :',X_LOW
read,PROMPT='Enter X HIGH :',X_HIGH
;X_HIGH=X_LOW+19.0
read,PROMPT='Enter Y LOW  :',Y_LOW
read,PROMPT='Enter Y HIGH :',Y_HIGH
;Y_HIGH=Y_LOW+19.0
read,PROMPT='Enter X:',X
read,PROMPT='Enter Y:',Y
```

```
close,1
!p.multi = 0
end
```

```
FUNCTION trial_fit,PET_IMAGEPATH_256,X_LOW,X_HIGH,Y_LOW,Y_HIGH
; Define the independent variable.
n = 20
x = FLOAT(INDGEN(20))
; Define the coefficients.
a = [1.0, 9.0, 0.5]
;print, 'Expected: For X Co-ordinate ', a
z = (x - a[1])/a[2]     ; Gaussian variable
!P.MULTI = [0,2,2]      ; set up 2x2 plot window
nterms=3
s=read_dicom(PET_IMAGEPATH_256)
;magnifiedImg = CONGRID(s, 614, 614, /INTERP)
;;C=intarr(564,566)
;C=intarr(512,512)
;;C=magnifiedImg[49:613,47:613]
;C=magnifiedImg[50:564,50:564]
b=fltarr(20)
k=0

for i=X_LOW,X_HIGH do begin
    for j=Y_LOW,Y_HIGH do begin
    b[k]=b[k]+s[i,j]
    endfor
k=k+1
endfor

y=b
;print,y
y = y + a[0]*exp(-z^2/2)
;print,y
; Fit the data to the function, storing coefficients in
; coeff:
yfit = GAUSSFIT(x, y, coeff, NTERMS=nterms)
;print, 'Result:FOR X   ', coeff[0:nterms-1]
; Plot the original data and the fitted curve:
;window, 0, xsize=800, ysize=500
;PLOT, x, y, TITLE='nterms='+STRTRIM(nterms,2),color=255*256L
;window, 1, xsize=800, ysize=400
;OPLOT, x, yfit, THICK=2

RETURN,coeff[1]+X_LOW
End
```

```
function MASK_spheres_3,x_low
a=x_low
point3=fltarr(7,7)
point3[0,0:6]=[0,0,0,0,0,0,0]
point3[1,0:6]=[0,0,0,123,141,120,0]
point3[2,0:6]=[0,0,145,261,316,209,117]
point3[3,0:6]=[0,107,193,451,590,356,134]
point3[4,0:6]=[0,0,170,398,531,331,134]
point3[5,0:6]=[0,0,120,202,258,193,116]
point3[6,0:6]=[0,0,0,117,134,115,0]
return,point3
END
```

```
function MASK_spheres_2,x_low
a=x_low
point2=fltarr(8,8)
point2[0,0:7]=[0,0,0,0,0,0,0,0]
point2[1,0:7]=[0,0,0,0,0,0,0,0]
point2[2,0:7]=[0,0,118,134,118,0,0,0]
point2[3,0:7]=[0,107,187,294,257,150,0,0]
point2[4,0:7]=[0,137,332,586,496,222,109,0]
point2[5,0:7]=[0,132,326,570,481,208,110,0]
point2[6,0:7]=[0,104,177,265,239,141,0,0]
point2[7,0:7]=[0,0,103,130,124,0,0,0]
return,point2
END
```

```
function MASK_spheres_1,x_low
a=x_low
point1=fltarr(6,6)
point1[0,0:5]=[0,0,0,0,0,0]
point1[1,0:5]=[101,0,0,0,0,0]
point1[2,0:5]=[168,160,118,0,0,0]
point1[3,0:5]=[410,396,190,108,0,0]
point1[4,0:5]=[558,548,241,106,0,0]
point1[5,0:5]=[331,331,176,0,0,0]
return,point1
END
```

```
function MASK_gam_SW,x_low
a=x_low
point=fltarr(16,16)
point[0,0:15] =[10,3,-5,-9,1,5,19,17,13,7,1,7,3,-2,-9,2]
point[1,0:15] =[2,-5,-10,3,19,21,18,13,16,13,15,19,14,5,-19,-1]
point[2,0:15] =[-11,-8,10,22,29,24,18,10,17,15,19,21,16,12,-3,-4]
point[3,0:15] =[-7,2,27,23,18,19,18,7,12,15,18,26,15,13,12,-1]
point[4,0:15] =[-7,16,30,15,11,26,26,13,18,19,18,23,22,21,18,8]
point[5,0:15] =[2,15,15,12,10,31,27,17,16,16,19,20,23,19,14,12]
point[6,0:15] =[14,14,14,20,14,20,6,6,4,-1,11,18,15,13,17,22]
point[7,0:15] =[7,22,27,24,19,14,-2,6,-1,-8,6,15,9,8,18,19]
point[8,0:15] =[10,26,20,21,25,24,8,11,-4,-3,15,25,24,17,20,14]
point[9,0:15] =[6,25,17,24,25,23,11,16,5,10,22,25,27,21,17,8]
point[10,0:15]=[-11,14,17,28,21,16,21,26,19,14,17,20,28,22,14,8]
point[11,0:15]=[-13,3,15,25,23,22,32,29,22,4,10,19,26,23,15,0]
point[12,0:15]=[-12,-7,5,21,29,22,22,22,19,5,18,21,14,17,14,-2]
point[13,0:15]=[-12,-1,-1,9,22,30,21,18,20,27,24,16,8,8,0,-8]
point[14,0:15]=[9,8,-3,-1,10,23,24,20,18,24,21,6,-5,-2,0,-2]
point[15,0:15]=[3,-2,-2,-2,-6,-7,-4,9,-1,0,3,-7,-13,-4,-5,0]
return,point
END
```

```
function MASK_gam_acr,x_low
a=x_low
point=fltarr(16,16)
point[0,0:15]=[7,6,15,33,68,106,131,131,129,127,104,73,54,23,2,0]
point[1,0:15]=[16,23,53,93,129,141,139,125,117,127,132,126,118,75,14,2]
point[2,0:15]=[22,56,103,125,140,137,127,130,126,129,136,133,134,117,63,
10]
point[3,0:15]=[34,95,126,131,130,125,126,132,129,131,128,129,123,119,110
,43]
point[4,0:15]=[64,127,129,129,135,135,125,112,107,122,129,129,110,112,13
3,79]
point[5,0:15]=[93,132,133,130,136,127,92,52,38,72,111,127,117,124,135,97
]
point[6,0:15]=[105,124,132,128,122,89,37,4,-3,27,78,121,126,128,132,113]
point[7,0:15]=[119,126,130,129,113,72,19,-4,-2,9,50,107,117,122,134,123]
point[8,0:15]=[111,124,131,137,121,91,23,-6,-2,8,55,110,116,129,134,124]
point[9,0:15]=[83,122,138,143,135,117,64,29,24,43,88,124,124,136,133,117
]
point[10,0:15]=[72l,130,137,130,125,122,119,101,88,104,118,123,122,136,1
26,91]
point[11,0:15]=[41,113,135,124,125,124,130,129,119,127,129,127,125,137,1
23,66]
point[12,0:15]=[15,71,126,133,126,128,125,130,126,129,123,125,129,134,97
,27]
point[13,0:15]=[12,34,86,118,130,133,129,137,138,124,112,128,126,100,37,
-2]
point[14,0:15]=[3,7,31,70,103,123,124,132,131,118,112,116,88,45,0,-6]
point[15,0:15]=[-1,0,-4,11,45,78,88,106,99,82,72,57,27,13,5,1]
return,point
END
```

VITA