Test on matRad_gammaIndex.m

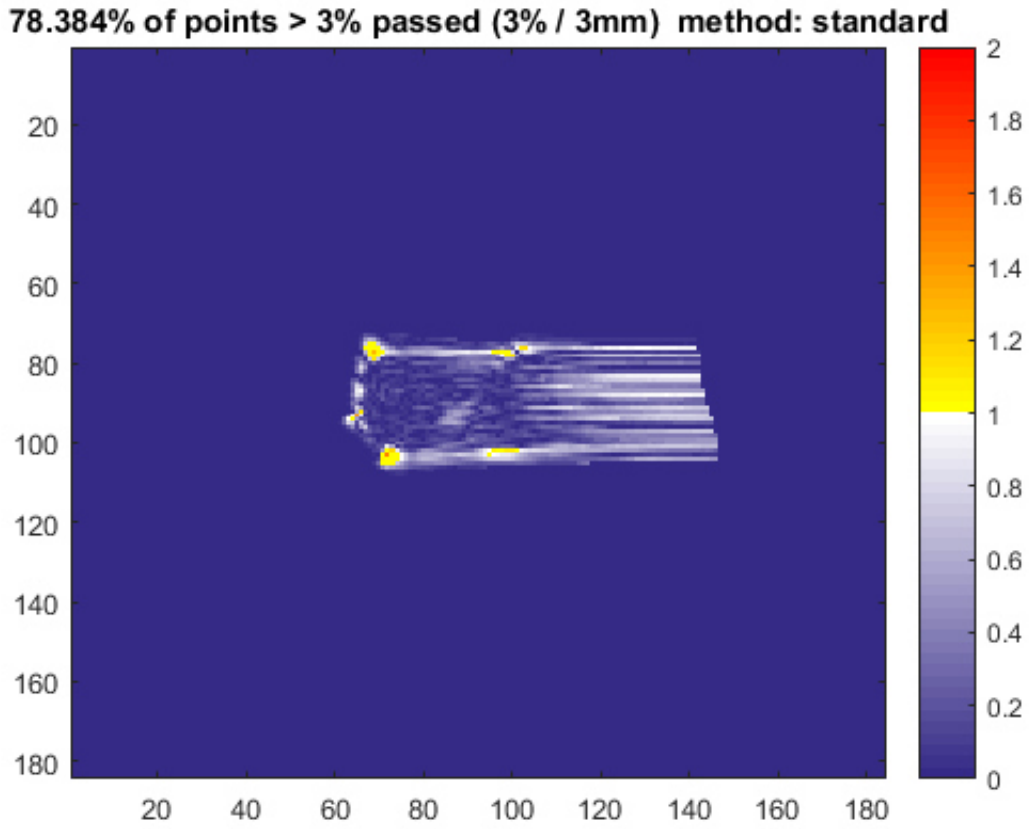
I run a test aim to understand the difference in the results of this function for different interpolation input methods and dimensions. I add a test on local and global gamma index calculation. I impose a threshold of 1% and 1mm

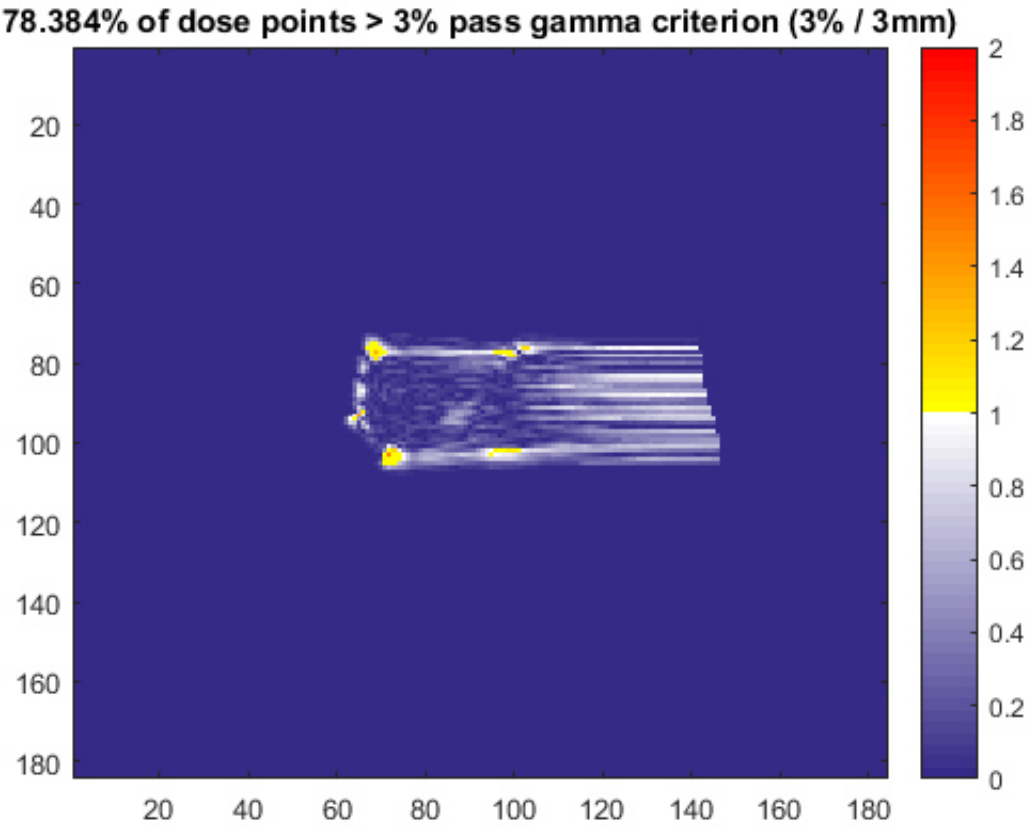
```
close all
threshold = [3 3];
```

Here we can see the differences between the new and the old programs

```
figure
tic;
[~,~,passrate_s] = matRad_gammaIndex_p(dose_5mm,dose_3mm,[ct.resolution.x ct.resolution.y
ct.resolution.z],...
    threshold,round(pln.isoCenter(1,3)/ct.resolution.z),'standard',0,'global');
time0=toc;

figure
matRad_gammaIndex_old(dose_5mm,dose_3mm,[ct.resolution.x ct.resolution.y ct.resolution.z],
...
    threshold,round(pln.isoCenter(1,3)/ct.resolution.z));
```

```
matRad: using gamme criteria 3[mm], 3[%]matRad: using gamme criteria 3[mm], 3[%].
.
.
.
.
```





Here we check passrates with linear interpolation figure tic; [~,~,passrate_l(1)] = matRad_gammaIndex_p(dose_5mm,dose_3mm,[ct.resolution.x ct.resolution.y ct.resolution.z],... threshold,round(pln.isoCenter(1,3)/ct.resolution.z),'linear',1,'global'); timeg(1)=toc; figure tic; [~,~,passrate_l(2)] = matRad_gammaIndex_p(dose_5mm,dose_3mm,[ct.resolution.x ct.resolution.y ct.resolution.z],... threshold,round(pln.isoCenter(1,3)/ct.resolution.z),'linear',2,'global'); timeg(2)=toc; figure tic; [~,~,passrate_l(3)] = matRad_gammaIndex_p(dose_5mm,dose_3mm,[ct.resolution.x ct.resolution.y ct.resolution.z],...

threshold,round(pln.isoCenter(1,3)/ct.resolution.z),'linear',3,'global'); timeg(3)=toc; figure tic; [~,~,passrate_l(4)] = matRad_gammaIndex_p(dose_5mm,dose_3mm,[ct.resolution.x ct.resolution.y ct.resolution.z],... threshold,round(pln.isoCenter(1,3)/ct.resolution.z),'linear',4,'global'); timeg(4)=toc;

figure subplot(1,2,1) plot([0:size(passrate_l,2)],[passrate_s passrate_l]) subplot(1,2,2) plot([0:size(timeg,2)],[time0 timeg])

%% % I repeat the same with cubic interpolation figure [~,~,passrate_c(1)] = matRad_gammaIndex_p(dose_5mm,dose_3mm,[ct.resolution.x ct.resolution.y ct.resolution.z],... threshold,round(pln.isoCenter(1,3)/ct.resolution.z),'cubic',1,'global'); figure [~,~,passrate_c(2)] = matRad_gammaIndex_p(dose_5mm,dose_3mm,[ct.resolution.x ct.resolution.y ct.resolution.z],... threshold,round(pln.isoCenter(1,3)/ct.resolution.z),'cubic',2,'global'); figure [~,~,passrate_c(3)] = matRad_gammaIndex_p(dose_5mm,dose_3mm,[ct.resolution.x ct.resolution.y ct.resolution.z],... threshold,round(pln.isoCenter(1,3)/ct.resolution.z),'cubic',3,'global'); figure [~,~,passrate_c(4)] = matRad_gammaIndex_p(dose_5mm,dose_3mm,[ct.resolution.x ct.resolution.y ct.resolution.z],... threshold,round(pln.isoCenter(1,3)/ct.resolution.z),'cubic',4,'global');

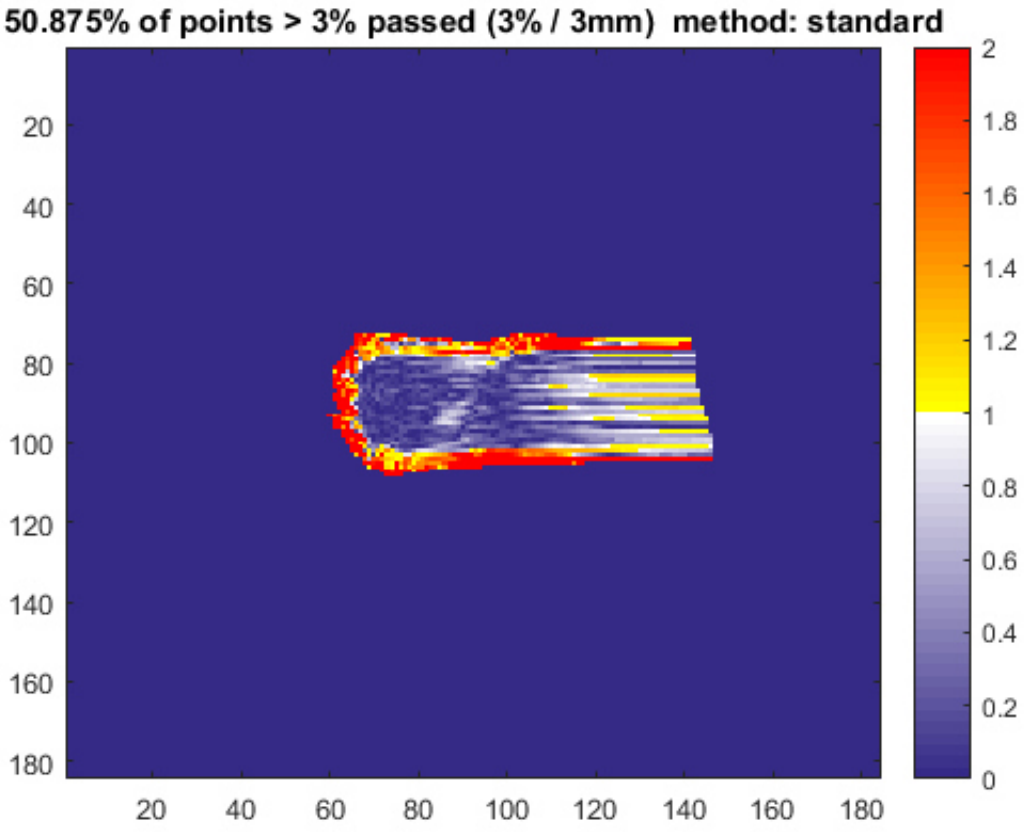
figure hold plot([0:size(passrate_l,2)],[passrate_s passrate_l],'b') plot([0:size(passrate_c,2)],[passrate_s passrate_c],'r') legend('linear','cubic')

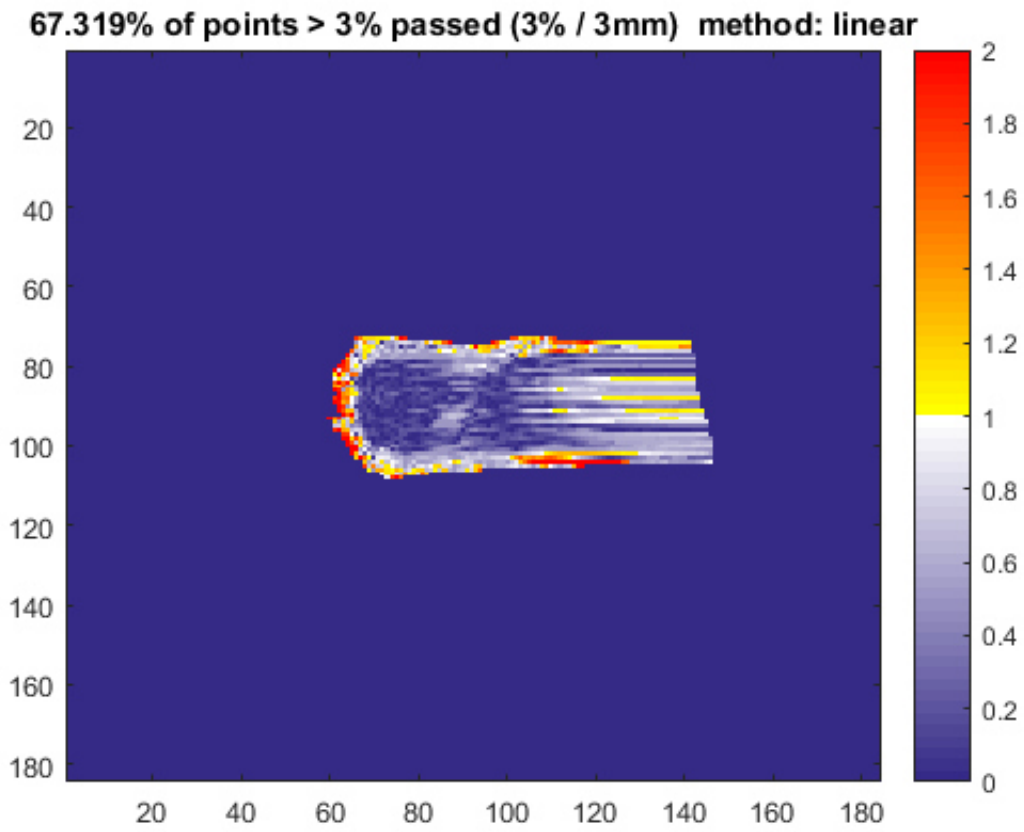
in this part we have the same results for local gamma calculation

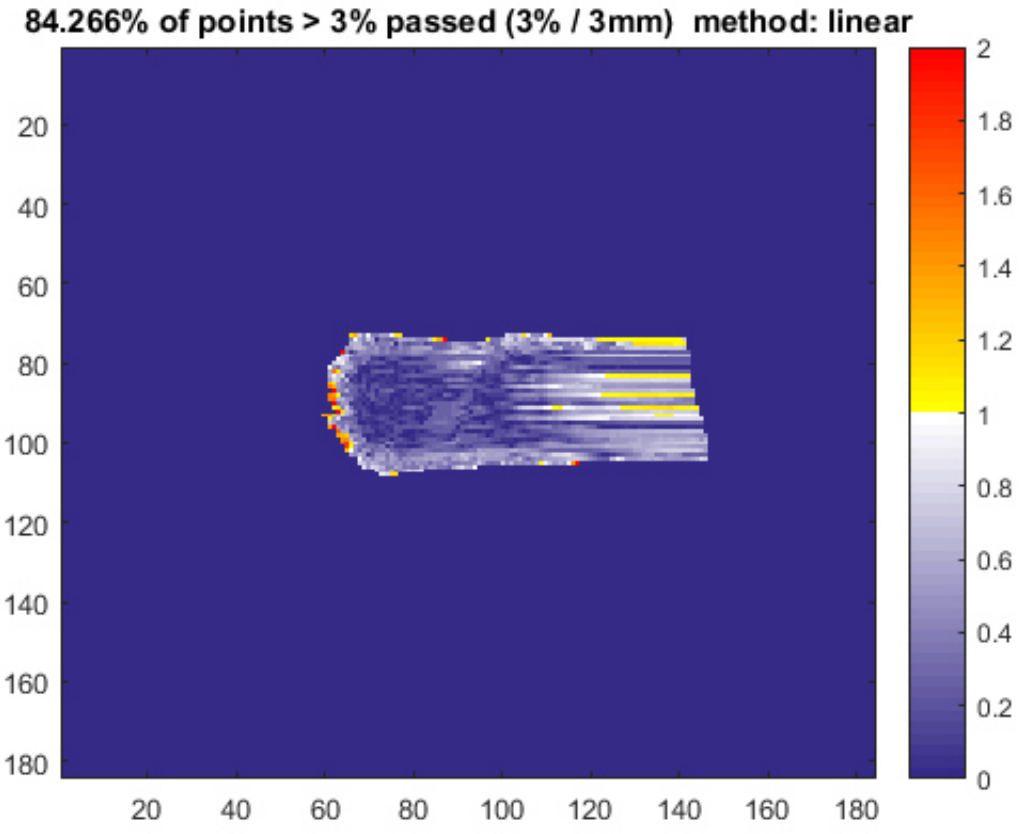
```
figure

[~,~,passrateloc_s] = matRad_gammaIndex_p(dose_5mm,dose_3mm,[ct.resolution.x ct.resolution
.y ct.resolution.z],...
    threshold,round(pln.isoCenter(1,3)/ct.resolution.z),'standard',0,'local');

figure
tic;
[~,~,passrateloc_l(1)] = matRad_gammaIndex_p(dose_5mm,dose_3mm,[ct.resolution.x ct.resolut
ion.y ct.resolution.z],...
    threshold,round(pln.isoCenter(1,3)/ct.resolution.z),'linear',1,'local');
timeg(1) = toc;

figure
tic;
[~,~,passrateloc_l(2)] = matRad_gammaIndex_p(dose_5mm,dose_3mm,[ct.resolution.x ct.resolut
ion.y ct.resolution.z],...
    threshold,round(pln.isoCenter(1,3)/ct.resolution.z),'linear',2,'local');
timeg(2) = toc;

figure
tic;
[~,~,passrateloc_l(3)] = matRad_gammaIndex_p(dose_5mm,dose_3mm,[ct.resolution.x ct.resolut
ion.y ct.resolution.z],...
    threshold,round(pln.isoCenter(1,3)/ct.resolution.z),'linear',3,'local');
timeg(3) = toc;

figure
tic;
[~,~,passrateloc_l(4)] = matRad_gammaIndex_p(dose_5mm,dose_3mm,[ct.resolution.x ct.resolut
ion.y ct.resolution.z],...
  threshold,round(pln.isoCenter(1,3)/ct.resolution.z),'linear',4,'local');
timeg(4) = toc;

figure
[~,~,passrateloc_c(1)] = matRad_gammaIndex_p(dose_5mm,dose_3mm,[ct.resolution.x ct.resolut
ion.y ct.resolution.z],...
```

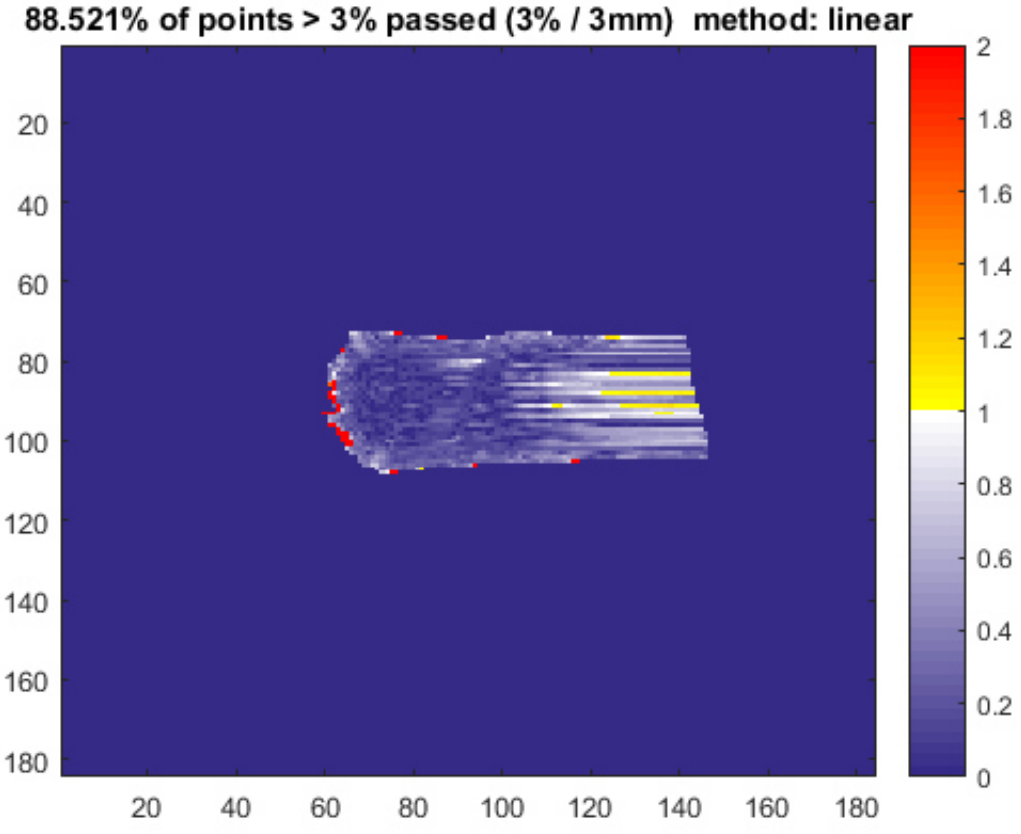
```
    threshold,round(pln.isoCenter(1,3)/ct.resolution.z),'cubic',1,'local');
figure
[~,~,passrateloc_c(2)] = matRad_gammaIndex_p(dose_5mm,dose_3mm,[ct.resolution.x ct.resolut
ion.y ct.resolution.z],...
    threshold,round(pln.isoCenter(1,3)/ct.resolution.z),'cubic',2,'local');
figure
[~,~,passrateloc_c(3)] = matRad_gammaIndex_p(dose_5mm,dose_3mm,[ct.resolution.x ct.resolut
ion.y ct.resolution.z],...
    threshold,round(pln.isoCenter(1,3)/ct.resolution.z),'cubic',3,'local');
figure
[~,~,passrateloc_c(4)] = matRad_gammaIndex_p(dose_5mm,dose_3mm,[ct.resolution.x ct.resolut
ion.y ct.resolution.z],...
    threshold,round(pln.isoCenter(1,3)/ct.resolution.z),'cubic',4,'local');

figure
subplot(1,2,1)
plot([0:size(passrateloc_l,2)],[passrateloc_s passrateloc_l])
subplot(1,2,2)
plot([0:size(timeg,2)],[time0 timeg])

figure
hold
plot([0:size(passrateloc_l,2)],[passrateloc_s passrateloc_l],'b')
plot([0:size(passrateloc_c,2)],[passrateloc_s passrateloc_c],'r')
legend('linear','cubic')
```

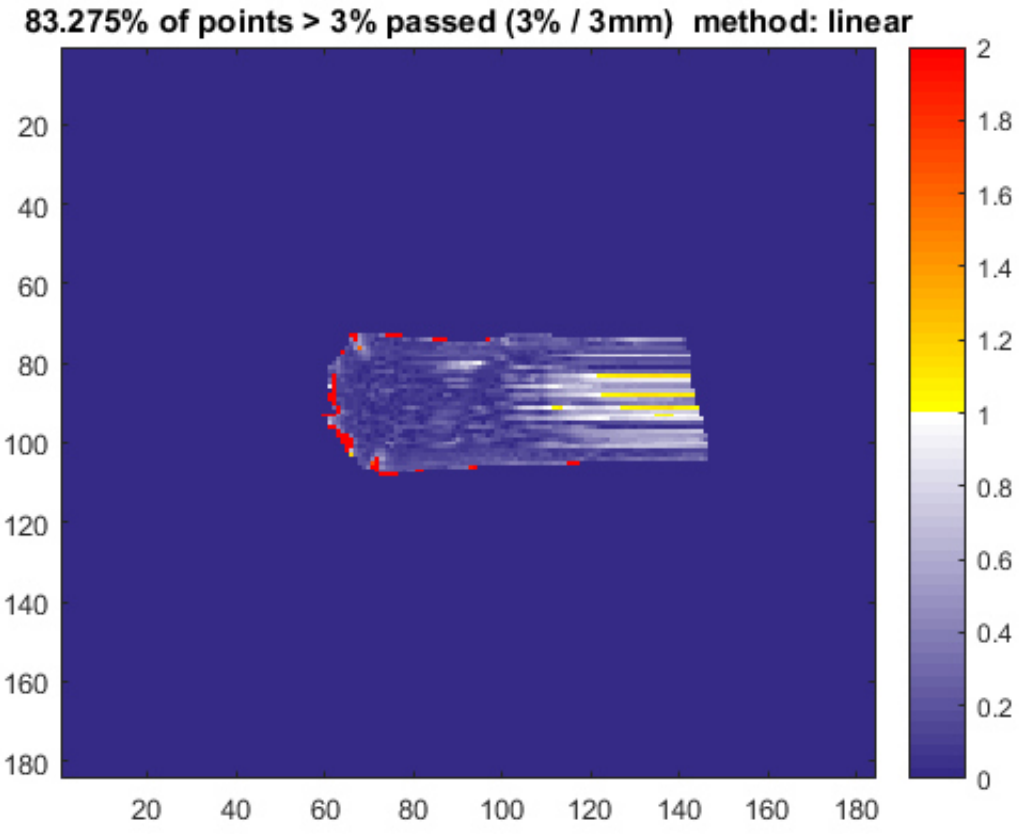
```
matRad: using gamme criteria 3[mm], 3[%]matRad: using gamme criteria 3[mm], 3[%]matRad: us
ing gamme criteria 3[mm], 3[%]matRad: using gamme criteria 3[mm], 3[%]matRad: using gamme
criteria 3[mm], 3[%]matRad: using gamme criteria 3[mm], 3[%]matRad: using gamme criteria 3
[mm], 3[%]matRad: using gamme criteria 3[mm], 3[%]matRad: using gamme criteria 3[mm], 3[%]
Current plot held
```

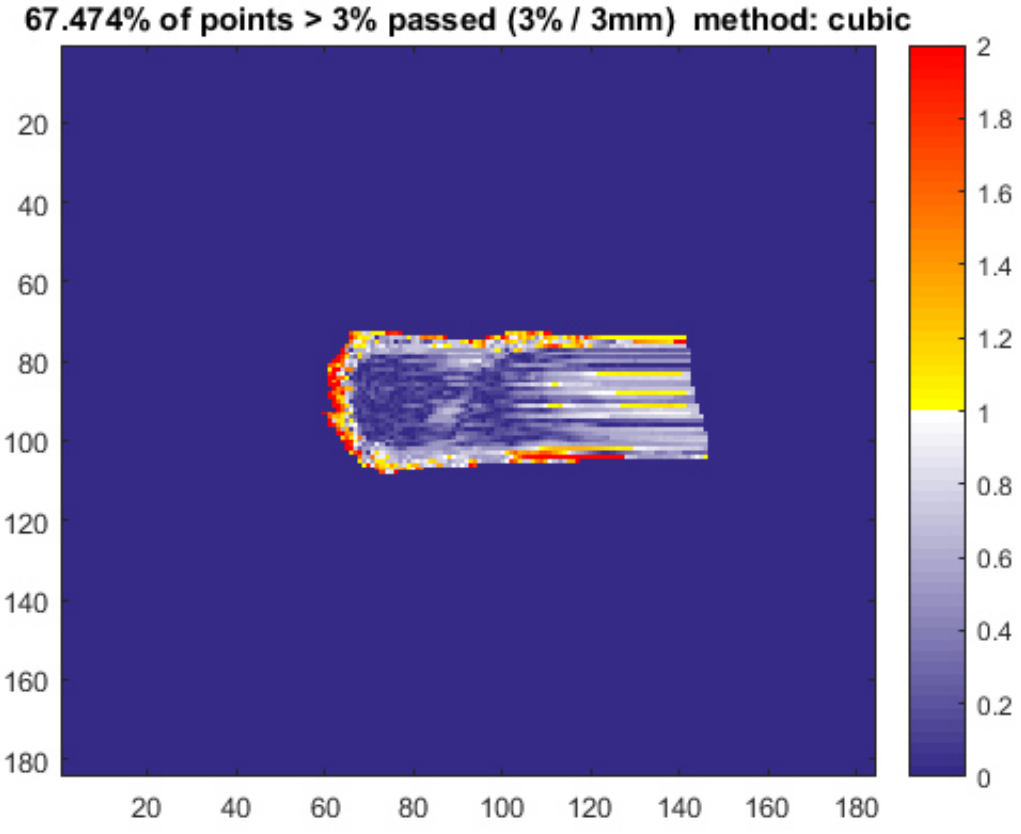

50.875% of points > 3% passed (3% / 3mm) method: standard
20
40
60
80
100
120
140
160
180
20
40
60
80
100
120
140
160
180
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0

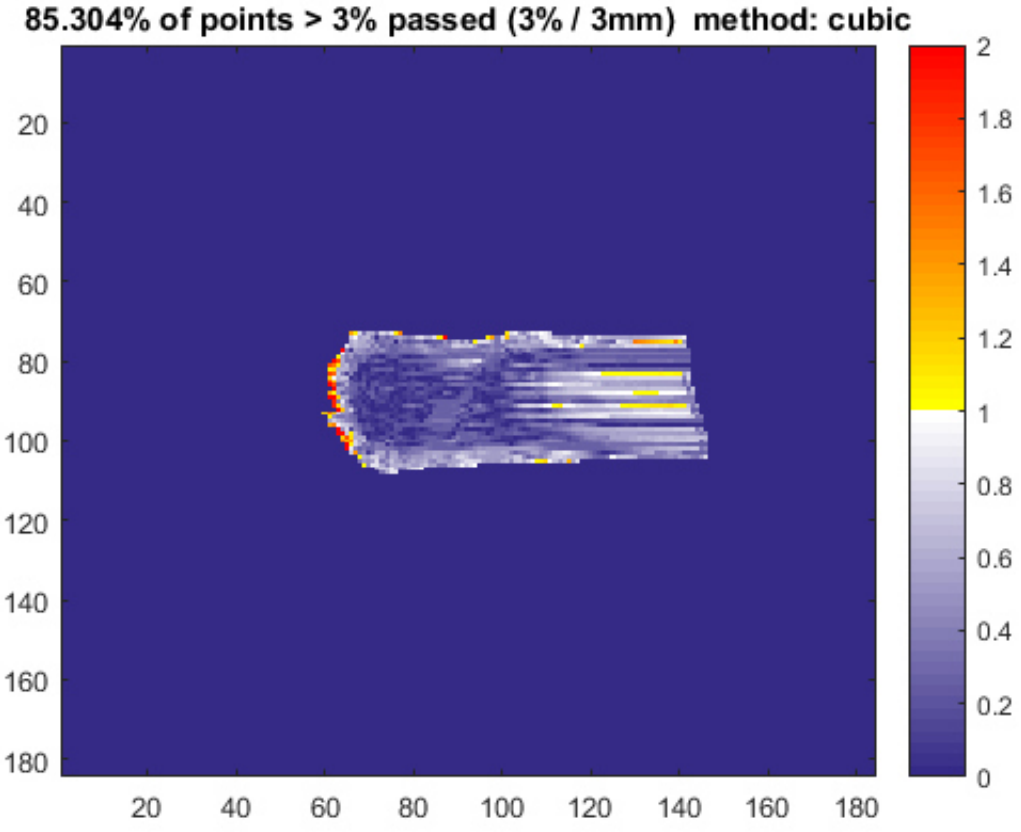

67.319% of points > 3% passed (3% / 3mm) method: linear
20
40
60
80
100
120
140
160
180
20
40
60
80
100
120
140
160
180
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0

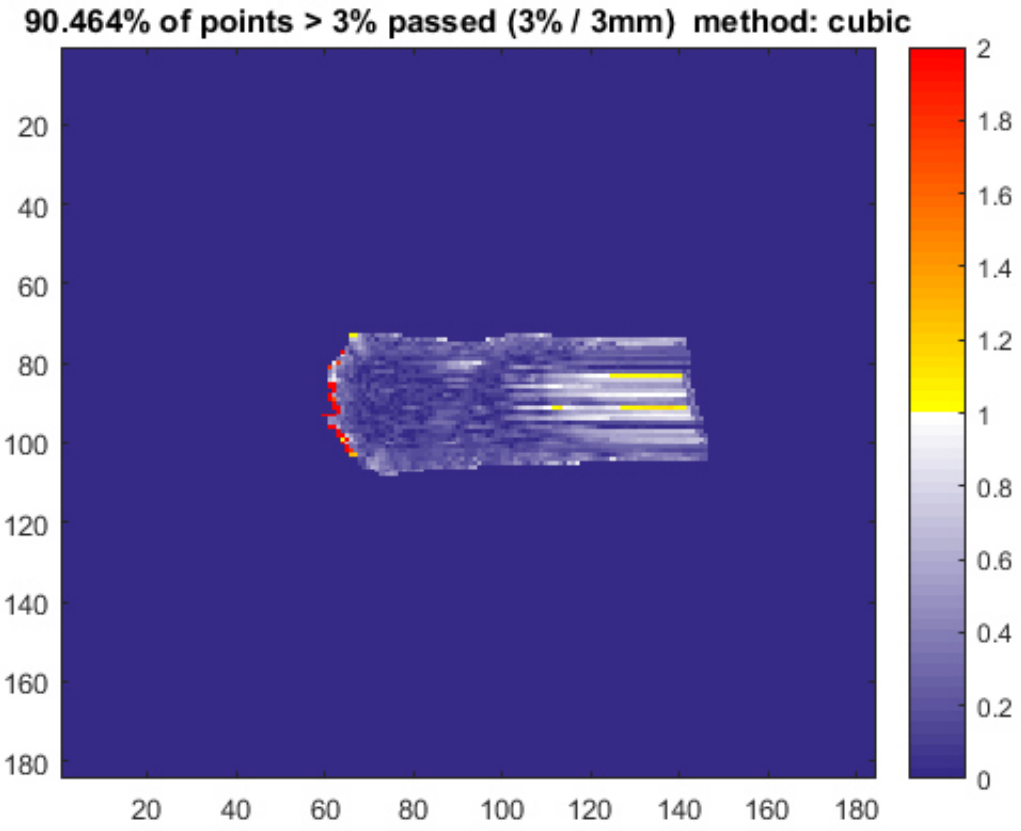

84.266% of points > 3% passed (3% / 3mm) method: linear
20
40
60
80
100
120
140
160
180
20 40 60 80 100 120 140 160 180
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0

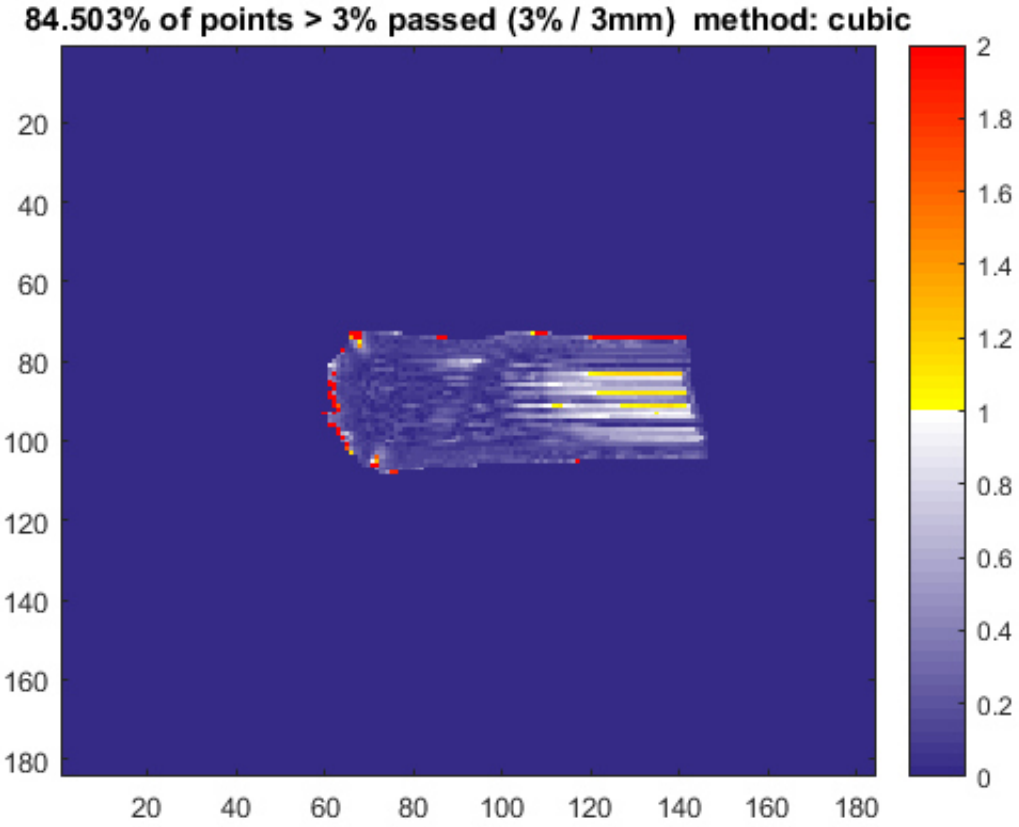

88.521% of points > 3% passed (3% / 3mm) method: linear
20
40
60
80
100
120
140
160
180
20 40 60 80 100 120 140 160 180
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0

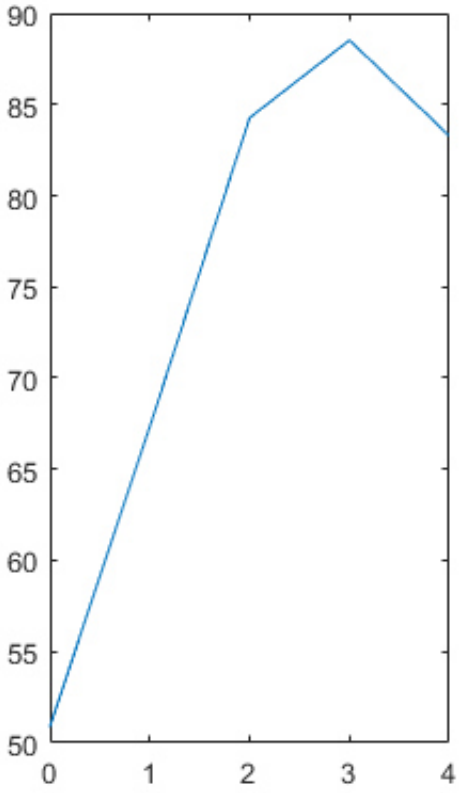

83.275% of points > 3% passed (3% / 3mm) method: linear
20
40
60
80
100
120
140
160
180
20
40
60
80
100
120
140
160
180
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0

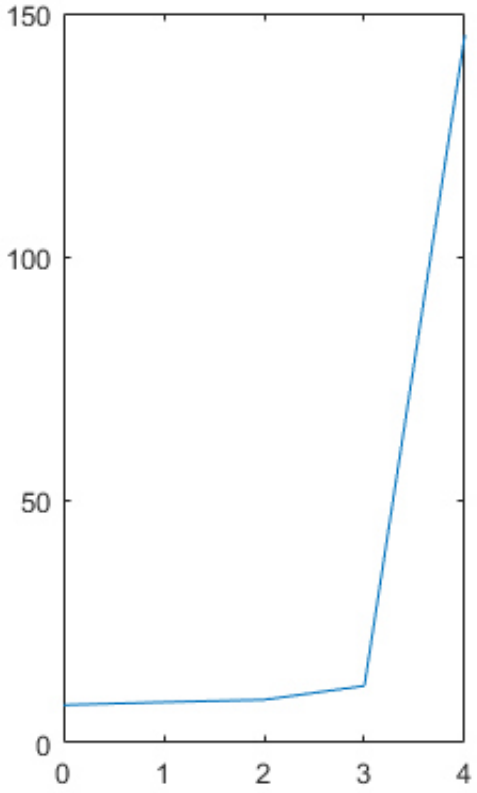

67.474% of points > 3% passed (3% / 3mm) method: cubic
20
40
60
80
100
120
140
160
180
20
40
60
80
100
120
140
160
180
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0

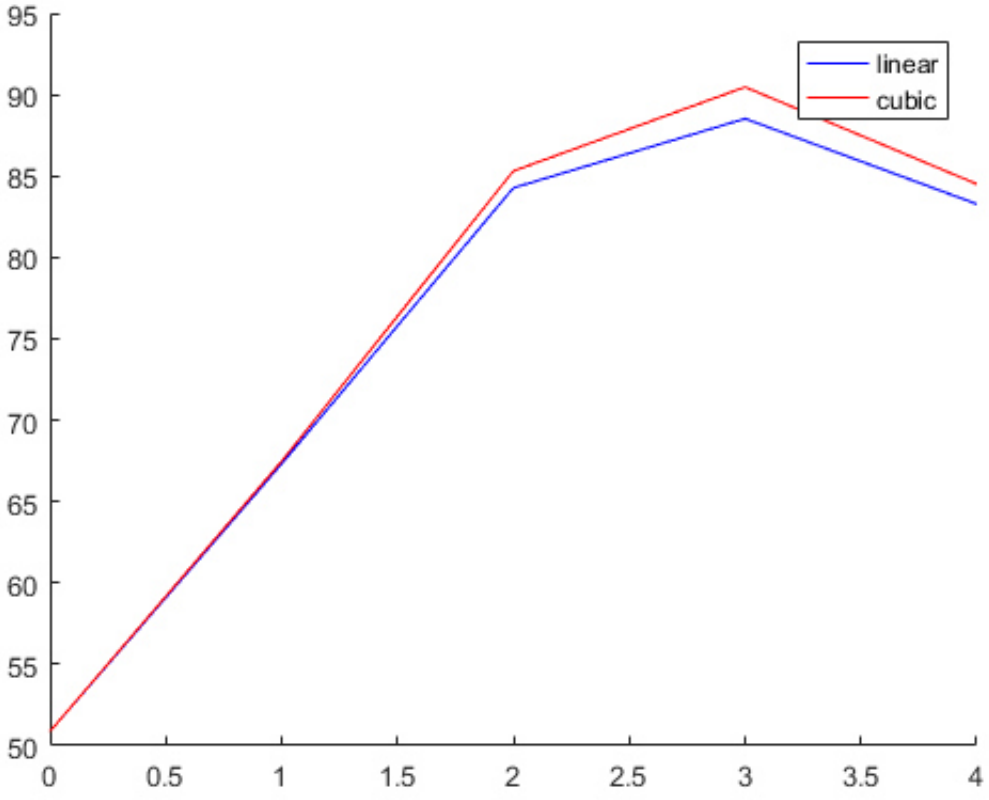

85.304% of points > 3% passed (3% / 3mm) method: cubic
20
40
60
80
100
120
140
160
180
20
40
60
80
100
120
140
160
180
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0


90.464% of points > 3% passed (3% / 3mm) method: cubic
20
40
60
80
100
120
140
160
180
20
40
60
80
100
120
140
160
180
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0


84.503% of points > 3% passed (3% / 3mm) method: cubic
20
40
60
80
100
120
140
160
180
20
40
60
80
100
120
140
160
180
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0


90
85
80
75
70
65
60
55
50
0
1
2
3
4


150
100
50
0
0
1
2
3
4



Published with MATLAB® R2016a