

Demonstration of CAMPS Algorithm with NCI Example CT images.

Introduction

This is the pixel attenuation algorithm based on CAMPS, intended to be used from publically available images from the NIH. This is derived from the CAMPS code used for the study. This is modified from the based on DICOM header values included in the sample images. This will display the DICOM header meta information. It will also calculate the SSDE based on the pixel attenuation process using the CAMPS morphologic approach.

To use this program, you must download the example images from the NCI NBIA website. The images are available from <https://imaging.nci.nih.gov/ncia/login.jsf>

I would suggest downloading individual images using the search function. If this implementation of the algorithm is used on more than 20 images, you will be waiting a long time for the program to execute. The extra time this program requires for execution is related to the display of images. Removing the recording of images from each slice, and the eventual display of the set of images, will dramatically reduce processing time (from about 20 minutes for 300 images, to about 6 minutes).

Try the following suggested steps for downloading individual images:

Select "Search Images".

Select in Collections "Head-Neck Cetuximab-"

Select "0522c001"

Look for Description "AbdomenCT 5.0 B40s"

Select "Thumbnails"

Scroll to page 10

Select "View Image" under 200

This depends on your browser, Select save or download, open download folder, copy downloaded dcm image to an empty folder you will use to contain a sequence of images.

Repeat the process for images 260,350.

the program is now ready to run.

`convfactor32[z_] := 3.704369 * Exp[-0.03671937 * z]; (* From AAPM Report 204*)`

```
ctimagedir = SystemDialogInput["Directory"];
testfiles = Select[FileNames["*.dcm", ctimagedir, Infinity], ! DirectoryQ[#] &];
```

(* This attempts to find DICOM files in the chosen file directory. The code searches any subdirectories. For my study I only transferred one sequence of axial images. Including multiple sequences would likely cause the algorithm to fail. This could be fixed by checking the DICOM header to include only primary axial sequences. For a spot check it is easy to instruct the radiation technologist to only send the axial sequences.*)

```
count = Length[testfiles];
```

```
dicomheader = Import[testfiles[[1]], "MetaInformation"];
dicomtags = <| |>;
AssociateTo[dicomtags, dicomheader];
dicomkeys = Keys[dicomtags];
```

```
(Print[#, "->", dicomtags[[#]]]) & /@ dicomkeys;
```

```
MetaElementGroupLength->196
```

```
FileMetaInformationVersion->{0, 1}
```

```
MediaStorageSOPClassUID->{1, 2, 840, 10008, 5, 1, 4, 1, 1, 2}
```

```
MediaStorageSOPInstanceUID->
```

```
{1, 3, 6, 1, 4, 1, 14519, 5, 2, 1, 5099, 8010, 126262270958899895316891821234}
```

```
TransferSyntaxUID->{1, 2, 840, 10008, 1, 2, 1}
```

```
ImplementationClassUID->{1, 2, 40, 0, 13, 1, 1, 1}
```

```
ImplementationVersionName->dcm4che-1.4.34
```

```
SpecificCharacterSet->ISO_IR 100
```

```
ImageType->{ORIGINAL, PRIMARY, AXIAL, CT_SOM5 SPI}
```

```
SOPClassUID->{1, 2, 840, 10008, 5, 1, 4, 1, 1, 2}
```

```
SOPInstanceUID->{1, 3, 6, 1, 4, 1, 14519, 5, 2, 1, 5099, 8010, 126262270958899895316891821234}
```

```
StudyDate->{1999, 8, 25}
```

```
SeriesDate->{1999, 8, 25}
```

```
AcquisitionDate->{1999, 8, 25}
```

```
ContentDate->{1999, 8, 25}
```

```
StudyTime->{16, 17, 56.695}
```

```
SeriesTime->{16, 22, 6.014}
```

```
AcquisitionTime->{16, 23, 14.5205}
```

```
ContentTime->{16, 23, 14.5205}
```

```
AccessionNumber->2819497684894126
```

```
Modality->CT
```

```
Manufacturer->SIEMENS
```

```
StationName->MEDPC
```

StudyDescription->Abdomen^AbdomenPETCT

SeriesDescription->AbdomenCT 5.0 B40s

ManufacturerModel->Emotion

(0009,0010)->SIEMENS CT VA1 DUMMY

PatientName->0522c0002

PatientID->0522c0002

PatientSex->M

PatientAge->{0, 0, 0}

PatientWeight->82.174

(0012,0010)->American_College_of_Radiology_Imaging_Network

(0012,0020)->4500

(0012,0021)->RTOG 0502 / ACRIN 4500

(0012,0042)->2

(0012,0062)->YES

(0012,0063)->Per DICOM PS 3.15 AnnexE. Details in 0012,0064

(0012,0064)->{{CodeValue -> 113100, CodingSchemeDesignator -> DCM,
CodeMeaning -> Basic Application Confidentiality Profile},
{CodeValue -> 113101, CodingSchemeDesignator -> DCM, CodeMeaning -> Clean Pixel Data Option},
{CodeValue -> 113103, CodingSchemeDesignator -> DCM, CodeMeaning -> Clean Graphics Option},
{CodeValue -> 113105, CodingSchemeDesignator -> DCM, CodeMeaning -> Clean Descriptors Option},
{CodeValue -> 113107, CodingSchemeDesignator -> DCM,
CodeMeaning -> Retain Longitudinal With Modified Dates Option}, {CodeValue -> 113108,
CodingSchemeDesignator -> DCM, CodeMeaning -> Retain Patient Characteristics Option},
{CodeValue -> 113109, CodingSchemeDesignator -> DCM, CodeMeaning -> Retain Device Identity Option},
{CodeValue -> 113111, CodingSchemeDesignator -> DCM, CodeMeaning -> Retain Safe Private Option}}}

BodyPartExamined->HEADNECK

SliceThickness->5

KVP->130

SoftwareVersions->VA40C

Protocol->AbdomenPETCT

ReconstructionDiameter->500

SourceDetectorDistance->940

SourcePatientDistance->535

GantryDetectorTilt->0

TableHeight->112

RotationDirection->CW

ExposureTime->800

XRaysTubeCurrent->140

Exposure->112
 FilterType->0
 GeneratorPower->18
 FocalSpots->0.95
 LastCalibrationDate->{1999, 8, 25}
 LastCalibrationTime->{9, 12, 28.}
 PatientPosition->HFS
 (0019,0010)->SIEMENS CT VA0 COAD
 StudyInstanceUID->{1, 3, 6, 1, 4, 1, 14 519, 5, 2, 1, 5099, 8010, 217 836 670 708 542 506 360 829 799 868}
 SeriesInstanceUID->{1, 3, 6, 1, 4, 1, 14 519, 5, 2, 1, 5099, 8010, 251 402 186 443 276 877 513 851 675 309}
 SeriesNumber->2
 AcquisitionNumber->2
 InstanceNumber->210
 ImagePosition->{-249.512, -361.512, -748.6}
 ImageOrientation->{1, 0, 0, 0, 1, 0}
 ReferenceFrameUID->{1, 3, 6, 1, 4, 1, 14 519, 5, 2, 1, 5099, 8010, 458 683 884 574 110 075 399 985 105 799}
 SliceLocation->-748.6
 (0021,0010)->SIEMENS MED
 SamplesPerPixel->1
 PhotometricInterpretation->MONOCHROME2
 Rows->512
 Columns->512
 PixelSpacing->{0.976563, 0.976563}
 AllocatedBits->16
 StoredBits->12
 HighBit->11
 PixelRepresentation->0
 BurnedInAnnotation->NO
 (0028,0303)->MODIFIED
 WindowCenter->{40, 300}
 WindowWidth->{300, 1500}
 RescaleIntercept->-1024
 RescaleSlope->1
 WindowGeometryExplanation->{WINDOW1, WINDOW2}
 (0029,0010)->SIEMENS CSA HEADER
 (0029,0011)->SIEMENS MEDCOM HEADER

RequestedProcedureDescription->Abdomen AbdomenPETCT

StorageMediaFilesetUID->

{1, 3, 6, 1, 4, 1, 14 519, 5, 2, 1, 5099, 8010, 133 499 539 961 587 437 118 090 881 675}

(* The DICOM header does not have to be displayed for every image. However, it is best to examine the DICOM header when applying this program to CT images from a previously untested CT machine. The program may need to be modified based on information included in the DICOM header. The calculation of SSDE in this algorithm is based on:

- a) the "ImageType" (which should be "ORIGINAL", "PRIMARY", "AXIAL"),
- b) the pixel spacing (for the area of the pixel),
- c) the rescale slope and intercept (needed to convert data back to Hounsfield units)
- d) the stored bits {to convert the stored integer data to the fractional pixel value used in the Mathematica algorithms}.

The DICOM header is displayed for the first NIH image. Note that the rescale slope is 1. The rescale intercept is -1024. The bits stored is 12. The image type is primary axial. These indicate that the tissue values should work, and that the images are axial. There is no slice CTDI vol, so I would read the reported CTDIvol from the dose summary page. There is no dose page available, so I will use a demonstration value (15mGy) for the CTDIvol.*)

ctdivol = 15;

uppertissuevalue = N[(1024 + 150) / 4096];

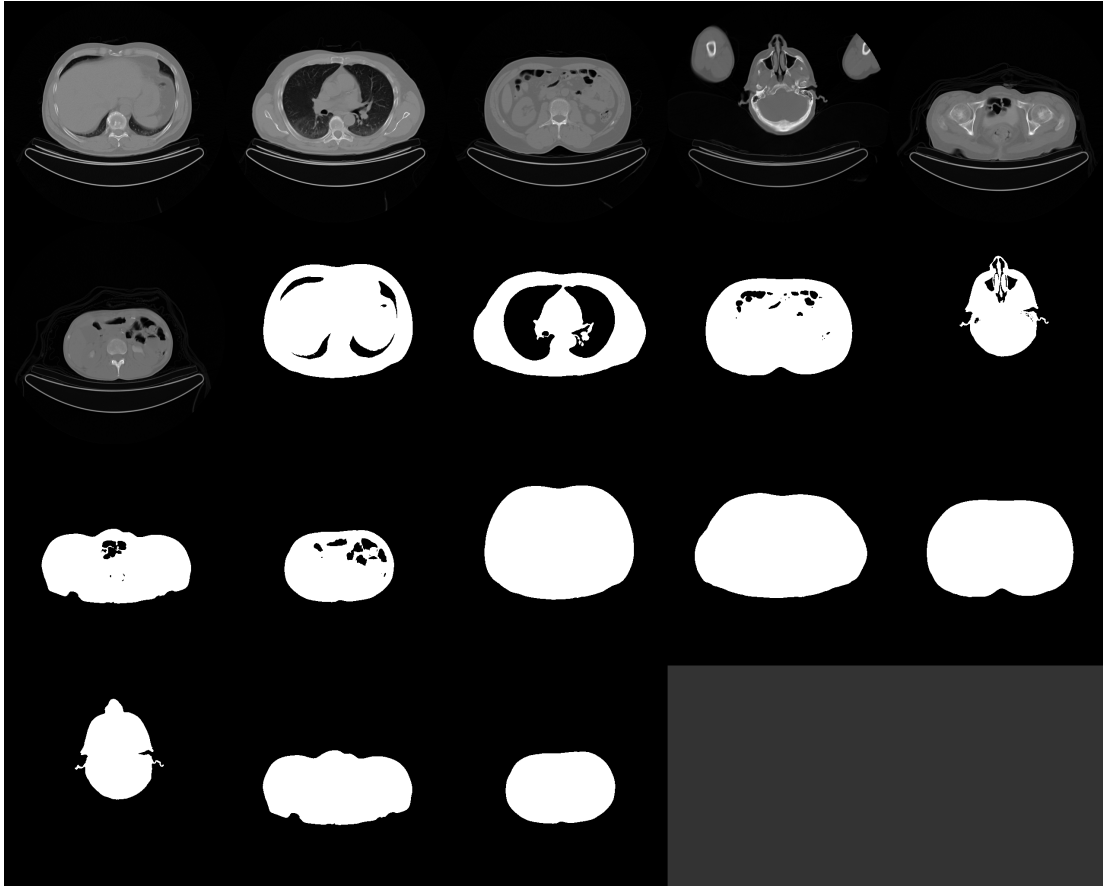
lowertissuevalue = N[(1024 - 300) / 4096];

```

stdimage = {};
morbinimage = {};
masks = {};
(dcmfile = #;
  dicomheader = Import[dcmfile, "MetaInformation"];
  dicomtags = <| |>;
  AssociateTo[dicomtags, dicomheader];
  dicomkeys = Keys[dicomtags];
  If[MemberQ[dicomkeys, "PixelSpacing"], {pixelspacingx, pixelspacingy} =
    dicomtags[["PixelSpacing"], {pixelspacingx, pixelspacingy} = {.97, .97}];
  If[MemberQ[dicomkeys, "(0018,9345)"], slicectdivol = dicomtags[["(0018,9345)"],
    slicectdivol = ctdivol];
  (* dicomtags[["(0018,9345)"]; DICOM tag for slice CTDivol*)
  imagetypes = dicomtags[["ImageType"]];
  If[! AnyTrue[imagetypes, StringMatchQ[#, "AXIAL"] &], Return];
  (* Skip non axial images... not yet tested flow control. *)
  im = Import[dcmfile];
  imb = DeleteSmallComponents[
    MorphologicalBinarize[im, {lowertissuevalue, uppertissuevalue}]];
  stdimage = Append[stdimage, ImageAdjust[im]];
  morbinimage = Append[morbinimage, imb];
  imd = Image[Import[dcmfile, "Data"][[1]]];
  imf = FillingTransform[imb];
  AppendTo[masks, imf];
) & /@ testfiles

Show[ImageCollage[Flatten[{stdimage, morbinimage, masks}]], ImageSize -> Large]
{Null, Null, Null, Null, Null, Null}

```



(* Note that this program is slowed by the display of multiple images. *)

```

(dcmfile = #;
  dicomheader = Import[dcmfile, "MetaInformation"];
  dicomtags = <| |>;
  AssociateTo[dicomtags, dicomheader];
  dicomkeys = Keys[dicomtags];
  If[MemberQ[dicomkeys, "PixelSpacing"], {pixelspacingx, pixelspacingy} =
    dicomtags[["PixelSpacing"]], {pixelspacingx, pixelspacingy} = {.97, .97}];
  If[MemberQ[dicomkeys, "(0018,9345)"], slicectdivol = dicomtags[["(0018,9345)"],
    slicectdivol = ctdivol];
  (* dicomtags[["(0018,9345)"]; DICOM tag for slice CTDivol*)
  imagetypes = dicomtags[["ImageType"]];
  If[! AnyTrue[imagetypes, StringMatchQ[#, "AXIAL"] &], Return];
  (* Skip non axial images... not yet tested flow control. *)
  im = Import[dcmfile];
  imb = DeleteSmallComponents[
    MorphologicalBinarize[im, {lowertissuevalue, uppertissuevalue}]];
  (* The simple CAMPS algorithm determines estimated phantom
    diameter with the single code line of:
    edw= 2*pixelspacingx *.1*Values[ComponentMeasurements[
      imb,"EquivalentDiskRadius"]][[1]];
  *)
  imd = Image[Import[dcmfile, "Data"]][[1]];
  imf = FillingTransform[imb];
  npixels = Values[ComponentMeasurements[imf, "Count"]][[1]];
  adjpixs = PixelValue[imd, imf] - 1024;
  totpixs = Total[adjpixs];
  pixelarea = pixelspacingx * pixelspacingy;
  roiarea = (pixelarea * npixels) / 100;
  eDw = 2 * Sqrt[Abs[(Mean[adjpixs] / 1000) + 1] * roiarea / Pi];
  slice32adjfactorDw = convfactor32[eDw];
  slicessdeDw = slice32adjfactorDw * slicectdivol;
  Print[slicessdeDw]
) & /@testfiles
18.369
20.5129
20.2254
27.3029
22.7182
26.2132
{Null, Null, Null, Null, Null, Null}

```

Summary

In this collection of CT images from the NCI NBIA, the image at the base of the skull would not be appropriate for application of SSDE. If you wish to process a larger set of images, commenting out the

code for display of images would be helpful. In the Mathematica format, that is listed as “In(192)”. That will significantly increase the speed of the calculation.