Subject: Re: Problem reading DICOM files with IDL
Posted by David Fanning on Mon, 28 Jul 2003 15:02:48 GMT
View Forum Message <> Reply to Message

Nuno Chichorro Ferreira writes:

> I apologize if this has already been answered before. I have been unable
> to read DICOM files with IDL 5.3/5.4, can someone help me?
>
> The files I want to read are in DICOM compressed format (JPEG), but IDL
> only reads the size of the image, not the data. I know that IDL does not
> support compressed files (at least the 5.3/5.4 versions), but the same
> happens with the uncompressed files I have created by reading them with
> a DICOM viewer - Osiris - and saving them without compression.
>
> Can anyone find an easy solution for the problem?
> Many thanks,
>
> Nuno
>
> P.S. - The following is the error message I get when reading the
> compressed DICOM files:
>
> IDL> img=read_dicom(file)
> % Compiled module: READ_DICOM.
> % Loaded DLM: DICOM.
> % IDLFFDICOM::READ: This file uses the following compression type for
> its image(s): JPEG Lossless, Non-Hierarchical, First Order Prediction
> (Process 14)
>
> % IDLFFDICOM::READ: ** Unable to handle compressed data!!! **
>
> % IDLFFDICOM::READ: This image has an undefined length and may not be
> read properly.

It would be more helpful to see the error messages
caused by reading the uncompressed files, since even
you admit IDL isn't going to read the compressed files.

Sigh...

Cheers,

David

--
David W. Fanning, Ph.D.
Fanning Software Consulting, Inc.

Phone: 970-221-0438, E-mail: david@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Problem reading DICOM files with IDL
Posted by Rick Towler on Mon, 28 Jul 2003 16:15:31 GMT
View Forum Message <> Reply to Message

"Nuno Chichorro Ferreira" wrote in message ...

> I apologize if this has already been answered before. I have been unable
> to read DICOM files with IDL 5.3/5.4, can someone help me?
>

> % IDLFFDICOM::READ: This file uses the following compression type for
> its image(s): JPEG Lossless, Non-Hierarchical, First Order Prediction
> (Process 14)

This format "JPEG Lossless, Non-Hierarchical, First Order Prediction" isn't supported by IDLffDICOM. For a full list of supported/unsupported types check the docs, specifically the section on "IDL DICOM v3.0 Conformance Summary"

-Rick

Subject: Re: Problem reading DICOM files with IDL
Posted by Nuno Ferreira on Tue, 29 Jul 2003 14:52:37 GMT
View Forum Message <> Reply to Message

David Fanning wrote:

> It would be more helpful to see the error messages
> caused by reading the uncompressed files, since even
> you admit IDL isn't going to read the compressed files.
>
> Sigh...

I was hoping that IDL versions more recent than 5.4 would be able to read DICOM compressed files. In that case, the error message could be useful.

Anyway, here is the error message I get with the uncompressed files created with the Osiris DICOM viewer:

IDL> img=read_dicom(file)
% Compiled module: READ_DICOM.

% Loaded DLM: DICOM.
READ_DICOM: The file D:\TEMP\xa0001.dcm is not in a supported DICOM format.

Not very helpful, I know...
Just to check, I also tried the commands to read the header info that David mentioned in another message, which of course didn't work since IDL cannot read the file...

```
IDL> thisObject = Obj_New('IDLffDicom', file)
IDL> thisObject -> DumpElements
IDL> modality = thisObject -> GetValue('0008'xL, '0060'xL)
IDL> Print, "Modality: ", *modality
% Pointer type required in this context: MODALITY.
% Execution halted at:  $MAIN$
```

Are you aware of a DICOM viewer that reads compressed DICOM files and creates uncompressed files compatible with IDL? Do you have other ideas?
Thanks in advance...

Nuno

Subject: Re: Problem reading DICOM files with IDL
Posted by Wonko[3] on Wed, 30 Jul 2003 13:38:00 GMT
View Forum Message <> Reply to Message

nuno@ibili.uc.pt (Nuno Ferreira) writes:

> Are you aware of a DICOM viewer that reads compressed DICOM files and
> creates uncompressed files compatible with IDL? Do you have other ideas?
> Thanks in advance...

The Dicom Toolkit (DCMTK) from Offis (http://dicom.offis.de/) can do this. These are many command line utilities, available for Win32, Linux and Solaris in binary format, and as well as source code. What you need is the dcmdjpeg utility.

```
wonko@weird wonko $ dcmdjpeg --help
$dcmtk: dcmdjpeg v3.5.2 2002-12-23 $

dcmdjpeg: Decode JPEG-compressed DICOM file
usage: dcmdjpeg [options] dcmfile-in dcmfile-out

parameters:
  dcmfile-in                  DICOM input filename to be converted
  dcmfile-out                 DICOM output filename
```

```
general options:
  -h      --help                print this help text and exit
          --version             print version information and exit
  -v      --verbose               verbose mode, print processing details
  -d      --debug               debug mode, print debug information
processing options:
  color space conversion options:
    +cp   --conv-photometric     convert if YCbCr photom. interpr. (default)
    +cl   --conv-lossy           convert YCbCr to RGB if lossy JPEG
    +ca   --conv-always          always convert YCbCr to RGB
    +cn   --conv-never           never convert color space
  planar configuration options:
    +pa   --planar-auto          automatically determine planar configuration
                                 from SOP class and color space (default)
    +px   --color-by-pixel       always store color-by-pixel
    +pl   --color-by-plane       always store color-by-plane
  SOP Instance UID options:
    +ud   --uid-default          keep same SOP Instance UID (default)
    +ua   --uid-always           always assign new UID
output options:
  output file format:
    +F    --write-file           write file format (default)
    -F    --write-dataset        write data set without file meta information
  output transfer syntax:
    +te   --write-xfer-little    write with explicit VR little endian (default)
    +tb   --write-xfer-big       write with explicit VR big endian TS
    +ti   --write-xfer-implicit  write with implicit VR little endian TS
  post-1993 value representations:
    +u    --enable-new-vr        enable support for new VRs (UN/UT) (default)
    -u    --disable-new-vr       disable support for new VRs, convert to OB
  group length encoding:
    +g=   --group-length-recalc  recalculate group lengths if present (default)
    +g    --group-length-create  always write with group length elements
    -g    --group-length-remove  always write without group length elements
  length encoding in sequences and items:
    +e    --length-explicit      write with explicit lengths (default)
    -e    --length-undefined     write with undefined lengths
  data set trailing padding (not with --write-dataset):
    -p=   --padding-retain       do not change padding
                                 (default if not --write-dataset)
    -p    --padding-off          no padding (implicit if --write-dataset)
    +p    --padding-create       [f]ile-pad [i]tem-pad: integer
                                 align file on multiple of f bytes
                                 and items on multiple of i bytes
```

Alex

--

Alex Schuster Wonko@wonkology.org PGP Key available

alex@pet.mpin-koeln.mpg.de