DCMTK - Bug #620

DCMJPEG writes SOF0 JPEG marker when compressing 8-bit images in extended sequential mode (DICOM CP 1447)

2015-01-27 11:54 - Marco Eichelberg

Status:	Closed		**Start date:**	2015-01-27
Priority:	Normal		**Due date:**	
Assignee:	Marco Eichelberg		**% Done:**	0%
Category:	Library		**Estimated time:**	0:00 hour
Target version:	3.6.2			
Module:	dcmjpeg		**Compiler:**	
Operating System:				

Description

The DICOM standard currently has two different transfer syntaxes that permit lossy JPEG compression of images with 8 bits/sample:

- JPEG Baseline (8 bit only)
- JPEG Extended Sequential (8 or 12 bit)

DICOM CP 1447 clarifies that an image with 8 bits/sample, when encoded with extended sequential transfer syntax, should always contain the SOF1 start of frame marker in the JPEG bitstream, which indicates the "extended sequential" mode, a true superset of the JPEG baseline mode. The IJG JPEG library, which is used in DCMJPEG, however, encodes the SOF0 marker (which indicated JPEG Baseline) both in JPEG Baseline and JPEG Extended Sequential mode. The reason is that the IJG library does not actually distinguish between these two modes and never uses the additional features permitted in "extended sequential" mode, other than supporting 12 bits/pixel.

In order to make the library DICOM compliant according to CP 1447, we would have to modify the 8-bit version of the IJG library such that the calling user can choose whether SOF0 or SOF1 be written during lossy compression. The calling JPEG encoder class would then have to pass the correct value depending on the DICOM transfer syntax, which is not "visible" on the IJG level. Note that it is valid to write a JPEG Baseline bitstream together with SOF1, since SOF1 only enables optional extensions.

Related issues:

Related to DCMTK - Conformance #663: CP-1447: "Clarify 8 bit in Extended JPEG...	**Closed**	**2015-09-18**

History

#1 - 2016-11-16 15:21 - Marco Eichelberg

- Assignee set to Marco Eichelberg

#2 - 2016-11-17 11:40 - Marco Eichelberg

- Status changed from New to Closed

Fixed by commit ~~ea56f13~~ 220efcc.