Looking for data in EXIF format

Asked 9 years, 3 months ago Modified 3 years, 9 months ago Viewed 7k times



I got the problem with my program made for downloading the DateTimeOrginal data from the JPG file. I found the document about it on the internet:



https://ExifTool.org/TagNames/EXIF.html



I see that the data I'm looking for is at 0x9003 address.



So right now what I'm trying to do is:



```
temp = fopen(name, "rb");
```

open the file binary

```
fseek (temp, 0x9003, SEEK_SET);
```

move the File pointer to the address

```
fscanf(temp, "%s", str);
```

and load the data to the char[] structure.

Is atleast any of that correct? I'm still thinking that i got the problem with the address, because after compile that program i see only some trash from the file.

```
c++ exif
```

Share Follow

```
edited Dec 12, 2019 at 18:56

StarGeek

4,968 2 19 31
```

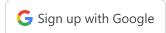
asked Jul 3, 2014 at 8:20



You are on the right track. You will need to verify that the data is compatible with a character string you are attempting to read the information into. I would suggest you download and look at the source for the program jhead. See: Exif Jpeg header manipulation tool It will provide an excellent example.

Join Stack Overflow to find the best answer to your technical question, help others answer theirs.

Sign up with email



Sign up with GitHub

Sign up with Facebook



This might be useful too: Read/Write exif/jpeg on windows using c++ - Ilya Jul 3, 2014 at 8:29

3 The data you're looking for starts with the 0x9003 *tag*. It's a magic number identifying the data that follows it. You're going to have to locate the Exif data block first, then locate the part of that block which starts with that tag, and that's where the data begins. – molbdnilo Jul 3, 2014 at 8:38

I'm still trying to find out how to calculate it. Could you give information, where to look for it? The data should begin in the same place for every picture? – General_Code Jul 3, 2014 at 8:54

2 Answers

Sorted by:

Highest score (default)





The EXIF data is embedded into the jpeg tag APP1 (0xE1).



The first thing to do is to find the jpef tag 0xE1 in the stream; you have to scan all the jpeg tags (marked by 0xFF+tag, in your case 0xFF,0xE1). After you get the tag, find its length by reading the next 2 bytes (and adjust for high endian), then get the tag's content.



After you get the tag's content, then look in it for the EXIF tag you are interested in (0x9003).



The method readStream in the jpeg class of the open source project Imebra gives you an example on how to parse jpeg tags:



https://bitbucket.org/binarno/imebra/src/2eb33b2170e76b5ad2737d1c2d81c1dcaccd19e5/project_files/library/imebra/src/jpegCodec.cpp?at=default#cl-867

Share Follow

answered Jul 3, 2014 at 8:50



Paolo Brandoli 4 679 26 38



Given the style of programming of the OP, I'd recommend Easyexif at https://github.com/mayanklahiri/easyexif It's relatively easy to integrate. Note that fseek() goes to a file position; it does not search for a certain number.



2

Share Follow

answered Dec 19, 2016 at 20:55



Ruud van Gaal 133 2 8



4

Join Stack Overflow to find the best answer to your technical question, help others answer theirs.

Sign up with email



X