LA-UR-18-20810
Approved for public release; distribution is unlimited.

Title: Extracting the Data From the LCM vk4 Formatted Output File

Author(s): Wendelberger, James G.

Intended for: Report

Issued: 2018-02-05

Extracting the Data From the LCM vk4 Formatted Output File

James Wendelberger
CCS-6
Los Alamos National Laboratory

29 January 2018

vk4 File Produced by Keyence VK Software

- This file is output by the Keyence laser conformal microscope (LCM)
- This file contains all of the data for a “run”
  - Height
  - Optical: Red Green Blue
  - Laser (Intensity)
  - Laser plus Optical
  - Run Parameters (rows, columns, etc)
- The file is a binary file
  - Reading the data correctly is needed for custom analysis

Custom Analysis

- Need pixel information
- Not directly available without multiple “clicks” in the Keyence software to output comma separated values (CSV) file format
- Convert to usable form from vk4 files
  - Fast
  - No clicking needed
  - Smaller file sizes for transfer
  - Easier to manage multiple (8) data types from the same collection run
    - Height – 1 image
    - Optical – 3 images Red, Green, Blue (RGB)
    - Optical plus Laser – 3 images Red, Green, Blue (RGB)
    - Intensity – 1 image

No Off the Shelf Way to Read the File

- Contacted Kevin Dugan of Keyence
- Suggested Work with Gwyddion to read the format
- Obtained partial format document from David Nečas (Yeti) yeti@gwyddion.net
- “it contains more or less the same information as the keyence module source code (which is public)“

Reading the Binary Data in a vk4 file

- od -tx4 -N80 -j00 TA_Stamp_20X_S1_Y1_X1.vk4 | more
- Output (Offset Line Number followed by Binary Data)

```
0000000          5f344b56          01020109          00000000          00000054
0000020          000002dc          002402f0          00480304          00000000
0000040          00000000          00600620          00000000          00000000
0000060          0090093c          00915af4          0092acac          0093fe64
0000100          0095501c          00000000          00000000          0095502c
```

- First hex number 5f344b56 is: TKV_
- Tenth hex number is 00600620

Various Offsets in Decimal Lines

- myoffset =
- Columns 1 through 6

  84 732 2360048 4719364 0 0
- Columns 7 through 12

  6293024 0 0 9439548 9526004 9612460
- Columns 13 through 18

  9698916 9785372 0 0 9785388 0

- Column 3: laser+color images
- Column 4: light intensity image
- Column 7: height image 0
- Column 10: CCD thumbnail images
- Etc.

Finding the Height image data

- Use the tenth hex number, 00600620, to find Ox1 + 2x16 + 6x256 + 0x4096 + 0x65536 + 6x1048576 = 32 + 1536 + 6291456 = 6293024
- Add 12 header hex numbers to this number, 6293024, equals 6293036 the number of hex numbers to skip to get to the laser image data
- Command: od -tx4 -N40 -j6293036 TA_Stamp_20X_S1_Y1_X1.vk4
- Output (Offset Line Numbers followed by Binary Data)

```
30003054    3233434d    00300000    00051681    0007b058
30003074    10000000    08200004    000c3000    50001040
30003114    18600014    001c7000
```

- char(hex2dec('4d')), char(hex2dec('43')), char(hex2dec('33')), char(hex2dec('32')) = “MC32”

Directly in MATLAB

- The commands

```
myfile='TA_Stamp_20X_S1_Y1_X1.vk4';
mydirectory='/Users/jgw/Documents/Documents/CCS-6 FILES/Crack Detection/Instrument Information/';
cd(mydirectory);
fileID = fopen(myfile);
A = fread(fileID,'ubit8');
myoffset=6293024
for myi=[1:3]
    i1=myoffset+1+(myi-1)*4;
    i2=i1+1;
    i3=i1+2;
    i4=i1+3;
    testoffset(myi)=A(i1)+A(i2)*256+A(i3)*256*256+A(i4)*256*256*256;
end

  testoffset(1:3)
  for myi=[4]
    i1=myoffset+1+(myi-1)*4;
    i2=i1+1;
    i3=i1+2;
    i4=i1+3;
    testoffset(myi)=A(i1)+A(i2)*256+A(i3)*256*256+A(i4)*256*256*256;
  end

  [char(A(i1)) char(A(i2)) char(A(i3)) char(A(i4))]
```

- hexdumpvk4

```
myoffset =

     6293024

ans =

     1024      768       32

ans =

'MC32'
```

Binary Output Beginning of Height Image Data

- Line number, Hex line number, 2 character binary output
- Underlined Values: 1024 and 768
- Next Values 32 and “MC32”

```
393315  00600620: 00 04 00 00 00 03 00 00 20 00 00 00 4D 43 33 32   ¿¿¿¿¿¿¿¿ ¿¿¿MC32
393316  00600630: 00 00 30 00 81 16 05 00 58 B0 07 00 00 00 00 10   ¿¿0¿¿¿¿X°¿¿¿¿¿¿
393317  00600640: 04 00 20 08 00 30 0C 00 40 10 00 50 14 00 60 18   ¿¿ ¿¿0¶¿@¿¿P¿¿`¿
393318  00600650: 00 70 1C 00 80 20 00 90 24 00 A0 28 00 B0 2C 00   ¿p¿¿¿ ¿¿ $¿ (¿°,¿
393319  00600660: C0 30 00 D0 34 00 E0 38 00 F0 3B 00 FC 3E 00 FF   À0¿Ð4¿à8¿ð;¿ü>¿ÿ
393320  00600670: 41 00 FF 44 00 FF 46 00 FF 48 00 FF 4A 00 FF 4C   A¿ÿD¿ÿF¿ÿH¿ÿJ¿ÿL
393321  00600680: 00 FF 4E 00 FF 50 00 FF 52 00 FF 54 00 FF 56 00   ¿ÿN¿ÿP¿ÿR¿ÿT¿ÿV¿
393322  00600690: FF 58 00 FF 5A 00 FF 5C 00 FF 5E 00 FF 60 00 FF   ÿX¿ÿZ¿ÿ\¿ÿ^¿ÿ`¿ÿ
393323  006006A0: 62 00 FF 64 00 FF 66 00 FF 68 00 FF 6A 00 FF 6C   b¿ÿd¿ÿf¿ÿh¿ÿj¿ÿl
393324  006006B0: 00 FF 6E 00 FF 70 00 FF 72 00 FF 74 00 FF 76 00   ¿ÿn¿ÿp¿ÿr¿ÿt¿ÿv¿
393325  006006C0: FF 78 00 FF 7A 00 FF 7C 00 FF 7E 00 FF 80 00 FF   ÿx¿ÿz¿ÿ|¿ÿ~¿ÿ¿¿ÿ
```

Color Image Information

- Color pixel info starts at: 147506

```
147504   004805E0: 00 00 04 00 00 00 00 00 00 00 03 00 00 00 00 00 18 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
147505   00480600: 00 00 00 00 24 00 00 00 43 00 6E 00 57 00 5D 00 ED 00 E2 00 3F 00 63 00 5F 00 15 00 1E 00 21 00
147506   00480620: 0C 00 10 00 14 00 08 00 0B 00 0D 00 10 00 14 00 19 00 22 00 28 00 32 00 08 00 09 00 0C 00 05 00
147507   00480640: 06 00 07 00 06 00 08 00 09 00 04 00 05 00 06 00 07 00 0A 00 09 00 12 00 19 00 19 00 12 00 1A 00
147508   00480660: 1A 00 06 00 09 00 09 00 06 00 09 00 09 00 07 00 09 00 08 00 07 00 09 00 07 00 0F 00 14 00 11 00
147509   00480680: 0D 00 12 00 0F 00 0F 00 14 00 10 00 0F 00 15 00 10 00 09 00 0D 00 0A 00 16 00 17 00 11 00 0E 00
```

- Skip first four 2 digit pairs then see
  - 1024
  - 768
  - 16
  - “WC16” [from char(hex2dec('31'))=1 and char(hex2dec('36'))=36]

Color Image Binary Data

```
147506  00480620: BA 6E 70 92 EB 63 45 21 75 14 EA 0D 74 19 90 32 4D 0C AB 07 A9 09 39 06 EC 0A 2F 19 7B 1A 47 09
147507  00480640: 94 09 61 09 EA 09 C7 14 9F 12 D4 14 CB 15 A2 0D 8F 17 AB 0E C8 07 62 05 A6 0A 0B 11 61 40 36 8F
147508  00480660: CC 74 FD 2E 1F 35 58 3A CB 37 3E 3A 4E 53 9D 78 B3 7B 7D 8F D0 7D 2B 7C 22 5D C3 27 B1 25 C9 52
147509  00480680: AF 94 9C 85 25 98 46 A6 77 96 EF 2F 2F 4A B9 5E F4 7B D4 82 8F 40 E6 79 35 96 8F 1F EB 19 21 10
147510  004806A0: 58 11 6C 28 D7 1E F2 12 91 07 DB 06 31 17 D4 78 7B 80 7F 27 F8 36 16 3B AB 33 A6 40 C1 3B 1E 5C
147511  004806C0: DD 3B 62 1B 40 18 4C 2D 4F 2C 7D 23 E2 3B B0 83 6C 66 15 15 A1 0B 1A 0D EC 0F 3F 0A 15 07 94 07
147512  004806E0: 28 06 CE 04 EC 04 60 06 65 09 3E 0F 08 0A 57 11 36 13 D9 09 C0 0E 81 15 B5 0D 5D 14 9D 28 E8 21
147513  00480700: 6C 0F 01 07 52 0A DC 22 3D 37 CD 1C 53 14 59 1F 16 1C F8 17 15 18 AC 0F 83 16 08 16 55 10 67 1A
147514  00480720: AE 3D 80 37 50 2B 40 39 7E 29 E7 45 B9 91 45 B0 39 B1 B9 82 B0 4A 66 2D 79 09 90 06 8D 07 5C 16
147515  00480740: 93 34 D1 20 05 15 04 08 60 06 BC 11 4C 0E 9A 08 3F 0E 98 0D F1 17 3F 5D E0 64 1C 1F 1E 14 5F 1E
147516  00480760: BA 1B 88 2B D6 2E 39 1A E6 23 D6 46 21 57 8D 45 81 31 A7 14 62 10 4A 26 C8 29 26 1B 45 31 58 2B
147517  00480780: 59 0A 41 0F BA 1B 43 10 7A 0A FC 0D 7F 0B 4A 07 2C 06 E5 07 68 07 EA 0B B8 18 68 20 BC 12 51 2D
147518  004807A0: F8 58 2A 31 29 14 4A 09 F0 08 4A 1C 81 19 51 08 A7 0A 01 27 3A 23 D2 51 1A 86 8A 64 9D 21 26 09
147519  004807C0: 60 07 0A 14 1F 26 43 49 72 8E 73 80 FA 72 70 C1 82 B5 AE 27 C1 12 57 0E E5 09 E4 0B 0A 35 1D 39
147520  004807E0: 54 0A C0 08 B3 19 86 1B B7 0A A1 09 9D 17 3C 19 F7 16 28 19 4D 12 9A 0B 95 0E FE 15 72 17 27 0F
147521  00480800: DC 0F 05 24 D3 4F 46 43 F5 39 5B 7B AD 58 24 30 F2 5D 08 CB C5 E3 F6 D5 18 69 06 13 CC 0F 54 39
147522  00480820: A1 4C 9E 4B D8 2E 65 0A 61 0F 8E 63 1E 82 DD 4D 74 25 A4 3F FD 6A 27 4B B9 0D B6 09 BF 14 3B 17
147523  00480840: 1F 11 85 13 1E 24 49 21 47 15 BE 22 85 A6 2B E7 B4 EE A3 ED 06 DC 25 83 10 83 FF BD 84 DE 2C EF
147524  00480860: 7A F9 43 F9 16 E9 F9 B5 4C 7F 26 6B FE 66 D2 78 50 7F DC 89 BA 8D A0 26 52 09 2A 07 CB 06 7B 07
147525  00480880: 6D 28 45 B4 81 D3 D8 BF 7B BD CE BE 96 BE 1A 86 C7 76 F9 A6 D6 9F 55 80 6B 77 37 5A 5F 2E CA 06
147526  004808A0: 9D 04 2D 05 75 03 1C 05 65 04 95 06 25 35 AD 37 76 20 75 46 AF 80 39 7E FF 7E 05 59 A8 2F B7 3F
147527  004808C0: 72 39 4C 22 EB 24 E8 1A 85 10 DA 18 91 56 AE 57 8F 39 FE 57 B1 48 BB 41 92 97 1A A1 61 61 8E 29
147528  004808E0: 81 36 94 4C 3D 46 D2 67 FA 96 61 9E 9A 3E 8D 22 C5 6C 34 B6 55 97 F9 34 A5 2C 89 33 92 29 98 14
147529  00480900: 0F 0D 3A 3D 5A 8E 3F 66 49 2C 19 63 1C 88 83 4B 2D 12 A0 20 87 78 E7 9F D0 93 E9 55 E1 1A 4F 19
147530  00480920: 6B 55 B0 80 8F 7B 9F 84 6C C4 78 A9 EA 4B 74 25 80 0F 90 15 E9 2A 6D 2A 53 27 2D 0D 3C 06 F9 08
147531  00480940: 3E 1C 82 46 D7 76 49 6E E9 1F 4F 09 5B 13 3D 7D C1 A5 9F 76 6C 46 1A 24 93 0B 4B 08 D7 0A D5 06
147532  00480960: B1 05 67 0B A8 10 C7 0A 4B 05 58 05 85 0C 06 10 1D 08 EA 05 78 05 4E 06 86 0C 43 0D 6A 09 8E 15
147533  00480980: 04 1E 38 12 84 08 D2 0A 73 0B 2A 0B 17 0C BF 0A 1E 2F BF 7E A9 9C 9F 8B F9 62 5C 93 6E D2 F9 C3
147534  004809A0: E4 88 0B 92 F2 8F 99 80 AD 71 33 3D 20 20 DA 10 C6 19 A1 3D 33 72 01 45 45 13 E4 18 59 32 2A 1E
147535  004809C0: 4B 26 8A 4B 13 47 0C 7B 5D B3 8C A9 A0 85 7F 6F D8 65 60 62 8F 42 43 35 2F 82 FD C4 0B B1 66 58
147536  004809E0: E3 16 A8 1A EE 4C 49 4B 10 54 FF 7F 59 9E 82 AD D1 A7 C9 86 90 5C 21 2B 3C 11 66 09 82 0B A0 11
147537  00480A00: 18 10 87 0F 75 14 3F 11 35 12 1E 12 EE 1B FA 4B A2 47 29 10 1B 0A C5 13 AC 0D 93 06 EC 06 73 09
147538  00480A20: FC 10 C0 16 34 13 77 0F 83 0A 51 0C D4 10 82 09 00 09 24 0F 5C 38 57 96 33 90 3D 21 2E 1A A3 46
```

Color Image Decimal and Binary Data

- Red Integers: 71 137 183 137 109 101 91 80 71 55
- Green Integers: 90 156 185 142 86 80 70 64 59 44
- Blue Integers 55 121 141 102 63 54 60 55 54 35
- Binary Start:

```
BA 6E 70 92 EB 63 45 21 75 14 EA 0D 74 19 90 32 4D 0C AB 07 A9 09 39 06 EC 0A 2F 19 7B 1A 47 09
94 09 61 09 EA 09 C7 14 9F 12 D4 14 CB 15 A2 0D 8F 17 AB 0E C8 07 62 05 A6 0A 0B 11 61 40 36 8F
```

Color Image

- Need to Decode
- LZW
- 24 = 16 + 8

Height Image Binary Data

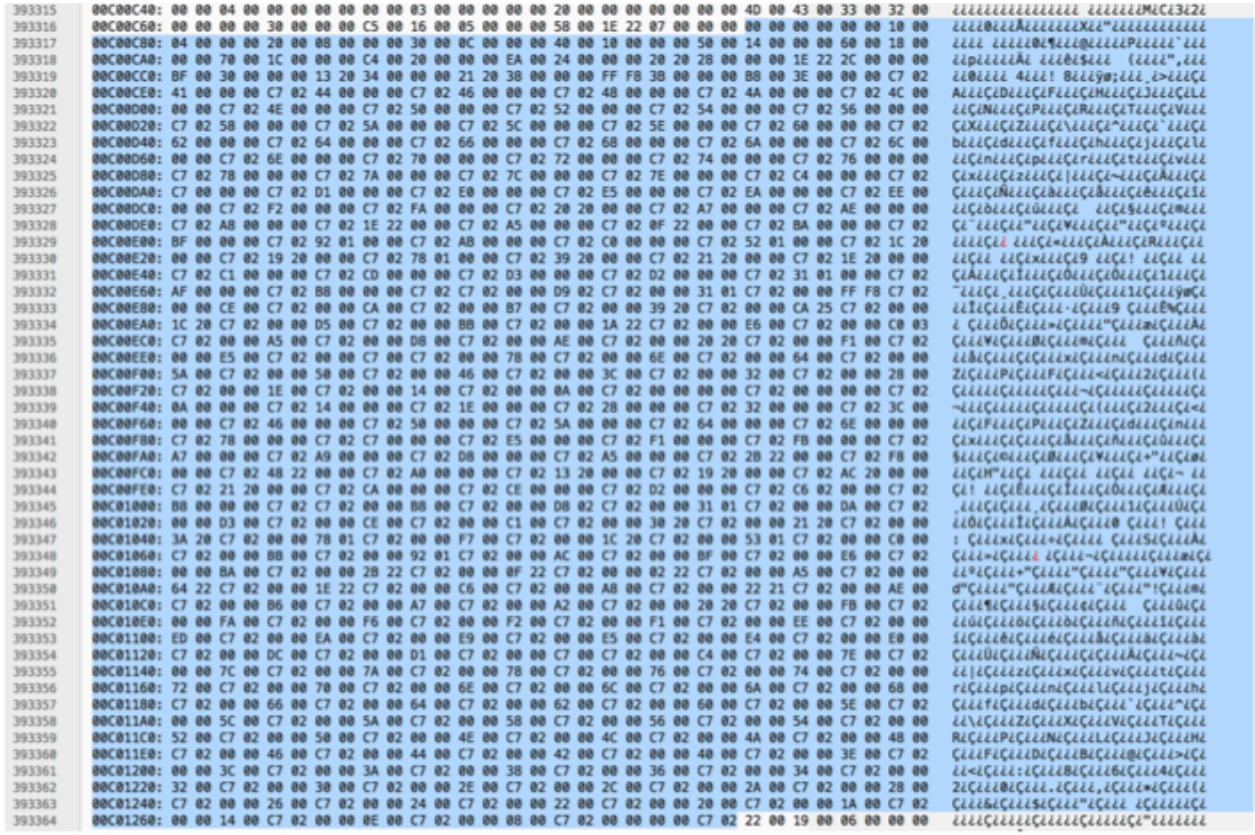
```
00600620: 00 04 00 00 00 03 00 00 20 00 00 00 4D 43 33 32 00 00 30 00 81 16 05 00 58 B0 07 00 00 00 00 10
00600640: 04 00 20 08 00 30 0C 00 40 10 00 50 14 00 60 18 00 70 1C 00 80 20 00 90 24 00 A0 28 00 B0 2C 00
00600660: C0 30 00 D0 34 00 E0 38 00 F0 3B 00 FC 3E 00 FF 41 00 FF 44 00 FF 46 00 FF 48 00 FF 4A 00 FF 4C
00600680: 00 FF 4E 00 FF 50 00 FF 52 00 FF 54 00 FF 56 00 FF 58 00 FF 5A 00 FF 5C 00 FF 5E 00 FF 60 00 FF
006006A0: 62 00 FF 64 00 FF 66 00 FF 68 00 FF 6A 00 FF 6C 00 FF 6E 00 FF 70 00 FF 72 00 FF 74 00 FF 76 00
006006C0: FF 78 00 FF 7A 00 FF 7C 00 FF 7E 00 FF 80 00 FF 82 00 FF 84 00 FF 88 00 FF 8C 00 FF 90 00 FF 94
006006E0: 00 FF 98 00 FF 9C 00 FF A0 00 FF A4 00 FF A8 00 FF AC 00 FF B0 00 FF B4 00 FF B8 00 FF BC 00 FF
00600700: C0 00 FF C4 00 FF C7 00 FF CB 00 FF CE 00 FF D2 00 FF D5 00 FF D9 00 FF DC 00 FF E0 00 FF E3 00
00600720: FF E7 00 FF EA 00 FF EE 00 FF F1 00 FF F5 00 FF F8 00 FF FC 00 FF FF 00 FA FF 00 F5 FF 00 F0 FF
00600740: 00 EB FF 00 E6 FF 00 E1 FF 00 DC FF 00 D7 FF 00 D2 FF 00 CD FF 00 C8 FF 00 C3 FF 00 BE FF 00 B9
00600760: FF 00 B4 FF 00 AF FF 00 A8 FF 00 A0 FF 00 96 FF 00 8C FF 00 82 FF 00 78 FF 00 6E FF 00 64 FF 00
00600780: 5A FF 00 50 FF 00 46 FF 00 3C FF 00 32 FF 00 28 FF 00 1E FF 00 14 FF 00 0A FF 00 00 FF 00 00 FF
006007A0: 0A 00 FF 14 00 FF 1E 00 FF 28 00 FF 32 00 FF 3C 00 FF 46 00 FF 50 00 FF 5A 00 FF 64 00 FF 6E 00
006007C0: FF 78 00 FF 82 00 FF 8C 00 FF 96 00 FF 9E 00 FF A4 00 FF A9 00 FF AF 00 FF B4 00 FF BA 00 FF BF
006007E0: 00 FF C5 00 FF CA 00 FF D0 00 FF D5 00 FF DB 00 FF E0 00 FF E6 00 FF EB 00 FF F1 00 FF F6 00 FF
00600800: FC 00 FF FF 00 FC FF 00 F9 FF 00 F5 FF 00 F2 FF 00 EE FF 00 EB FF 00 E7 FF 00 E4 FF 00 E0 FF 00
00600820: DD FF 00 D9 FF 00 D6 FF 00 D2 FF 00 CF FF 00 CB FF 00 C8 FF 00 C4 FF 00 C2 FF 00 C0 FF 00 BE FF
00600840: 00 BC FF 00 BA FF 00 B8 FF 00 B6 FF 00 B4 FF 00 B2 FF 00 B0 FF 00 AE FF 00 AC FF 00 AA FF 00 A8
00600860: FF 00 A6 FF 00 A4 FF 00 A2 FF 00 A0 FF 00 9E FF 00 9C FF 00 9A FF 00 98 FF 00 96 FF 00 94 FF 00
00600880: 92 FF 00 90 FF 00 8E FF 00 8C FF 00 8A FF 00 88 FF 00 86 FF 00 84 FF 00 82 FF 00 80 FF 00 7E FF
006008A0: 00 7C FF 00 7A FF 00 78 FF 00 76 FF 00 74 FF 00 72 FF 00 70 FF 00 6E FF 00 6C FF 00 6A FF 00 68
006008C0: FF 00 66 FF 00 64 FF 00 62 FF 00 60 FF 00 5E FF 00 5C FF 00 5A FF 00 58 FF 00 56 FF 00 54 FF 00
006008E0: 52 FF 00 50 FF 00 4E FF 00 4C FF 00 4A FF 00 48 FF 00 46 FF 00 44 FF 00 42 FF 00 40 FF 00 3E FF
00600900: 00 3C FF 00 3A FF 00 38 FF 00 36 FF 00 34 FF 00 32 FF 00 30 FF 00 2E FF 00 2C FF 00 2A FF 00 28
00600920: FF 00 26 FF 00 24 FF 00 22 FF 00 20 FF 00 1A FF 00 14 FF 00 0E FF 00 08 FF 00 00 FF 22 19 06 00
00600940: 42 03 06 00 E6 F0 05 00 D6 C9 05 00 4C 2B 06 00 88 9E 06 00 4C 2B 06 00 22 19 06 00 E6 F0 05 00
00600960: 22 19 06 00 4C 2B 06 00 A0 1D 06 00 AC 66 06 00 22 19 06 00 22 19 06 00 22 19 06 00 24 21 06 00
00600980: 88 9E 06 00 2A 52 06 00 88 9E 06 00 2A 52 06 00 72 78 06 00 C2 8C 06 00 88 9E 06 00 2A 52 06 00
006009A0: 4C 2B 06 00 2A 52 06 00 72 78 06 00 7E DA 06 00 7E DA 06 00 2A 52 06 00 2A 52 06 00 00 40 06 00
006009C0: 72 78 06 00 00 40 06 00 C2 8C 06 00 00 40 06 00 C2 8C 06 00 2A 52 06 00 00 40 06 00 2A 52 06 00
006009E0: AC 66 06 00 AC 66 06 00 4C 2B 06 00 00 40 06 00 C2 8C 06 00 7E DA 06 00 C2 8C 06 00 C2 8C 06 00
00600A00: 72 78 06 00 2A 52 06 00 C2 8C 06 00 00 40 06 00 00 40 06 00 C2 8C 06 00 2A 52 06 00 72 78 06 00
00600A20: AC 66 06 00 00 40 06 00 72 78 06 00 00 40 06 00 00 40 06 00 72 78 06 00 00 40 06 00 4C 2B 06 00
00600A40: 2A 52 06 00 AC 66 06 00 AC 66 06 00 72 78 06 00 72 78 06 00 72 78 06 00 2A 52 06 00 AC 66 06 00
00600A60: 00 40 06 00 72 78 06 00 AC 66 06 00 AC 66 06 00 C2 8C 06 00 C2 8C 06 00 2A 52 06 00 00 40 06 00
00600A80: C2 8C 06 00 C2 8C 06 00 72 78 06 00 00 40 06 00 88 9E 06 00 72 78 06 00 2A 52 06 00 00 40 06 00
00600AA0: 00 40 06 00 76 EC 06 00 88 9E 06 00 2A 52 06 00 72 78 06 00 72 78 06 00 72 78 06 00 C2 8C 06 00
00600AC0: C2 8C 06 00 8A 74 06 00 8A 5B 06 00 62 6A 06 00 00 40 06 00 72 78 06 00 7E DA 06 00 C2 8C 06 00
00600AE0: C2 8C 06 00 2A 52 06 00 2A 52 06 00 C2 8C 06 00 3C B3 06 00 72 78 06 00 00 40 06 00 AC 66 06 00
00600B00: AC 66 06 00 F8 6A 06 00 72 78 06 00 00 40 06 00 00 40 06 00 4C 2B 06 00 22 19 06 00 00 40 06 00
00600B20: 4C 2B 06 00 22 19 06 00 4C 2B 06 00 00 DC 05 00 5C A3 05 00 00 40 06 00 2A 52 06 00 2A 52 06 00
00600B40: AC 66 06 00 72 78 06 00 2A 52 06 00 AC 66 06 00 2A 52 06 00 2A 52 06 00 2A 52 06 00 00 40 06 00
```

Height Image First Binary Lines

- Values: 1024, 768, 32, MC32, 48,

```
393315    00C00C40: 00 00 04 00 00 00 00 00 00 00 03 00 00 00 00 00 20 00 00 00 00 00 00 00 4D 00 43 00 33 00 32 00
393316    00C00C60: 00 00 00 00 30 00 00 00 C5 00 16 00 05 00 00 00 58 00 1E 22 07 00 00 00 00 00 00 00 00 00 10 00
```

Height Image Binary Lines LZW Decoding Table



Bingo!!

- pn1706518:Instrument Information jgw$ od -N3145744 -j6293820 -D TA_Stamp_20X_S1_Y1_X1.vk4 | strings | head
- 30004474 399650 394050 389350 379350
- 30004514 404300 433800 404300 399650
- 30004534 389350 399650 404300 400800
- 30004554 419500 399650 399650 399650
- 30004574 401700 433800 414250 433800
- 30004614 414250 424050 429250 433800
- 30004634 414250 404300 414250 424050
- 30004654 449150 449150 414250 414250
- 30004674 409600 424050 409600 429250
- 30004714 409600 429250 414250 409600

Even Simpler!

```
pn1706518:Instrument Information jgw$ od   -N3145728  -j6293820  -
D   TA_Stamp_20X_S1_Y1_X1.vk4 | head
30004474            399650          394050          389350          379350
30004514            404300          433800          404300          399650
30004534            389350          399650          404300          400800
30004554            419500          399650          399650          399650
30004574            401700          433800          414250          433800
30004614            414250          424050          429250          433800
30004634            414250          404300          414250          424050
30004654            449150          449150          414250          414250
30004674            409600          424050          409600          429250
30004714            409600          429250          414250          409600

pn1706518:Instrument Information jgw$ od   -N3145728  -j6293820  -
D   TA_Stamp_20X_S1_Y1_X1.vk4 > simpler.txt
```

```
pn1706518:Instrument Information jgw$ od   -N3145728  -j6293820 -An  -
vtu4    TA_Stamp_20X_S1_Y1_X1.vk4 | head
                  399650          394050          389350          379350
                  404300          433800          404300          399650
                  389350          399650          404300          400800
                  419500          399650          399650          399650
                  401700          433800          414250          433800
                  414250          424050          429250          433800
                  414250          404300          414250          424050
                  449150          449150          414250          414250
                  409600          424050          409600          429250
                  409600          429250          414250          409600
pn1706518:Instrument Information jgw$ od   -N3145728  -j6293820 -An  -
vtu4    TA_Stamp_20X_S1_Y1_X1.vk4 | tail
                  433800          424050          414250          433800
                  443750          433800          359550          384000
                  404300          404300          409600          394050
                  414250          409600          389350          404300
                  374000          394050          399650          404300
                  394050          379350          349450          379350
                  419500          419500          419500          424050
                  424050          414250          439100          429250
                  409600          419500          419500          414250
```

```
pn1706518:Instrument Information jgw$ od   -N3145728  -j6293820 -An  -
vtu4    TA_Stamp_20X_S1_Y1_X1.vk4 | wc -l
  196609
pn1706518:Instrument Information jgw$ od   -N3145728  -j6293820 -An  -
vtu4    TA_Stamp_20X_S1_Y1_X1.vk4 | tail -1


pn1706518:Instrument Information jgw$ od   -N3145728  -j6293820 -An  -
vtu4    TA_Stamp_20X_S1_Y1_X1.vk4 | tail -2
                409600          419500          419500          414250


pn1706518:Instrument Information jgw$
```

>> 1024*768ans = 786432

>> 196609*4ans = 786436

One blank line of 4

Works!

MATLAB Code to Read vk4 File

```
myfile='TA_Stamp_20X_S1_Y1_X1.vk4';
mydirectory='/Users/jgw/Documents/Documents/CCS-6 FILES/Crack Detection/Instrument Information/';

cd(mydirectory);

mystring='VK4';

fileID = fopen(myfile);
% A = fread(fileID,'ubit8');
A = fread(fileID,'ubit8');
B = dec2hex(A);
char(A(1:20))'
char(A(end-62:end)');
char(B(1:20,1:size(B,2)))'
char(B(end-62:end)');

for myioffset=[1:18]
    i1=12+4*(myioffset-1)+1;
    i2=i1+1;
    i3=i1+2;
    i4=i1+3;
   myoffset(myioffset)=16*hex2dec(B(i1,1))+1*hex2dec(B(i1,2))+256*(16*hex2dec(B(i2,1))…
+1*hex2dec(B(i2,2)))+256*256*(16*hex2dec(B(i3,1))+1*hex2dec(B(i3,2)))+256*256*256*(16*hex2dec(B(i4,1))+1*hex2dec(B(i4,2)));
end
myoffset
A(6293023:6293084)';
A(6293821:6293884,1)'
C=A(6293821:9439548,1)';
D=dec2hex(C);
uncoded=  hex2dec(D(4,1))*16^7 +hex2dec(D(4,2))*16^6 +hex2dec(D(3,1))*16^5 + hex2dec(D(3,2))*16^4…
 + hex2dec(D(2,1))*16^3 + hex2dec(D(2,2))*16^2 + hex2dec(D(1,1))*16^1 + hex2dec(D(1,2))*16^0
F=reshape(C,4,1024*768)';
Fbase=[1 16^2 16^4 16^6]';
myheight=F*Fbase;
myheight=reshape(myheight,1024,768);
image(myheight/max(max(myheight)),'CDataMapping','scaled')
colorbar
image(myheight/max(max(myheight)),'CDataMapping','scaled')
colorbar
daspect([1 1 1])

myheight(1:8)
myheight(end-8:end)
```

```
clear all
myimagedirectory='/Users/jgw/Documents/Documents/CCS-6 FILES/Crack Detection/HCL_Crack /HCL_Pol_no_O2_Sec1_All/';
cd(myimagedirectory);
dnamelist = dir([myimagedirectory '*vk4']);
dnamelist([dnamelist.isdir]) = [];
totalimagefiles=length(dnamelist)
for myi=[1:totalimagefiles]
myimages{myi}=dnamelist(myi).name
end

[choice idx] = chooseimagedialog(myimages)
myfile=myimages{idx};


mystring='VK4';

fileID = fopen(myfile);
A = fread(fileID,'ubit8');
fclose(fileID);

char(A(1:20))'
char(A(end-62:end)');

Fbase=[1 16^2 16^4 16^6]';
for myioffset=[1:18]
   i1=12+4*(myioffset-1)+1;
   myoffset(myioffset)=A(i1:i1+3)'*Fbase;
end
myoffset

C=A(myoffset(7)+797:myoffset(10))';
rows=A(myoffset(7)+1:myoffset(7)+4)'*Fbase;
cols=A(myoffset(7)+5:myoffset(7)+8)'*Fbase;
codingbase=A(myoffset(7)+9:myoffset(7)+12)'*Fbase;
largestheightallowed=A(myoffset(7)+29:myoffset(7)+32)'*Fbase
F=reshape(C,4,rows*cols)';
myheight=F*Fbase;
myheight=reshape(myheight,rows,cols);

image(myheight/max(max(myheight)),'CDataMapping','scaled');
colorbar;
daspect([1 1 1]);

myheight(1:8)
myheight(end-8:end)
% %
% % min(min(myheight))
% % max(max(myheight))
```

Matlab Code to Read vk4 file: Choose a File

```
myimagedirectory='/Users/jgw/Documents/Docu
ments/CCS-6 FILES/Crack Detection/HCL_Crack
/HCL_Pol_no_O2_Sec1_All/';
cd(myimagedirectory);
dnamelist = dir([myimagedirectory'*vk4']);
dnamelist([dnamelist.isdir]) = [];
totalimagefiles=length(dnamelist)
formyi=[1:totalimagefiles]
myimages{myi}=dnamelist(myi).name;
end

[choice idx] = chooseimagedialog(myimages)
myfile=myimages{idx};
```

Matlab Code to Read vk4 file: Read the File

```
tic
mystring='VK4';

fileID = fopen(myfile);
A = fread(fileID,'ubit8');
fclose(fileID);

fileID = fopen(myfile);
AA = fread(fileID,'ubit8');
fclose(fileID);

char(A(1:20))'
char(A(end-62:end)');
```

Matlab Code to Read vk4 file: Compute Offsets

```
Fbase=[1 16^2 16^4 16^6]';
formyioffset=[1:18]
   i1=12+4*(myioffset-1)+1;
   myoffset(myioffset)=A(i1:i1+3)'*Fbase;
end
myoffset
```

Matlab Code to Read vk4 file: Read Optical Image

```
% set equal to 2 for optical data
% set equal to 3 for laser optical data
% set equal to 4 for laser intensity data
% set equal to 7 for height data
typeoffset=2;

opticalendoffset=min(myoffset((myoffset-myoffset(typeoffset))>0));
C=A(myoffset(typeoffset)+21:opticalendoffset)';
opticalrows=A(myoffset(typeoffset)+1:myoffset(typeoffset)+4)'*Fbase
opticalcols=A(myoffset(typeoffset)+5:myoffset(typeoffset)+8)'*Fbase
opticalcodingbase=A(myoffset(typeoffset)+9:myoffset(typeoffset)+12)'*Fbase
largestopticalallowed=A(myoffset(typeoffset)+29:myoffset(typeoffset)+32)'*Fbas
e
OF=reshape(C,3,opticalrows*opticalcols)';
myoptical=OF;
myoptical=reshape(myoptical,[opticalrows opticalcols 3]);
```

Matlab Code to Read vk4 file: Laser Optical Image

```
% set equal to 2 for optical data
% set equal to 3 for laser optical data
% set equal to 4 for laser intensity data
% set equal to 7 for height data
typeoffset=3;

colorendoffset=min(myoffset((myoffset-myoffset(typeoffset))>0));
C=A(myoffset(typeoffset)+21:colorendoffset)';
colorrows=A(myoffset(typeoffset)+1:myoffset(typeoffset)+4)'*Fbase
colorcols=A(myoffset(typeoffset)+5:myoffset(typeoffset)+8)'*Fbase
colorcodingbase=A(myoffset(typeoffset)+9:myoffset(typeoffset)+12)'*Fbase
largestcolorallowed=A(myoffset(typeoffset)+29:myoffset(typeoffset)+32)'*Fbas
e
CF=reshape(C,3,colorrows*colorcols)';
mycolors=CF;
mycolors=reshape(mycolors,[colorrows colorcols 3]);
```

Matlab Code to Read vk4 file: Read and Compute Laser Intensity Image

```
% set equal to 2 for optical data
% set equal to 3 for laser optical data
% set equal to 4 for laser intensity data
% set equal to 7 for height data
typeoffset=4;
%next offset and reshape only the image matrix
intensityendoffset=min(myoffset((myoffset-myoffset(typeoffset))>0));
%%%%%% works for height
C=A(myoffset(typeoffset)+797:intensityendoffset)';

intensityrows=A(myoffset(typeoffset)+1:myoffset(typeoffset)+4)'*Fbase
intensitycols=A(myoffset(typeoffset)+5:myoffset(typeoffset)+8)'*Fbase
intensitycodingbase=A(myoffset(typeoffset)+9:myoffset(typeoffset)+12)'*Fbase
largestintensityallowed=A(myoffset(typeoffset)+29:myoffset(typeoffset)+32)'*Fbase

IF=reshape(C,2,intensityrows*intensitycols)';

Fbase2=[1 16^2 ]';
myintensity=IF*Fbase2;
symsy(x);
checkintensity=round(fmyintensity(myintensity));
myintensity=reshape(checkintensity,intensityrows,intensitycols);
```

Matlab Code to Read vk4 file: Read Height Image

```
% set equal to 2 for optical data
% set equal to 3 for laser optical data
% set equal to 4 for laser intensity data
% set equal to 7 for height data
typeoffset=7;
%next offset and reshape only the image matrix
heightendoffset=min(myoffset((myoffset-myoffset(typeoffset))>0));

C=A(myoffset(typeoffset)+797:heightendoffset)';

heightrows=A(myoffset(typeoffset)+1:myoffset(typeoffset)+4)'*Fbase
heightcols=A(myoffset(typeoffset)+5:myoffset(typeoffset)+8)'*Fbase
heightcodingbase=A(myoffset(typeoffset)+9:myoffset(typeoffset)+12)'*Fbase
largestheightallowed=A(myoffset(typeoffset)+29:myoffset(typeoffset)+32)'*Fbas
e

HF=reshape(C,4,heightrows*heightcols)';
myheight=HF*Fbase;
myheight=reshape(myheight,heightrows,heightcols);
```

Matlab Code to Read vk4 file: Timing

```
preimagestime=toc
```

Matlab Code to Read vk4 file: Display Height Image

```
figure(4)

image(myheight’/max(max(myheight)),'CDataMapping','scaled');

colormap('jet')
colorbar;
daspect([1 1 1]);
set(gcf,'Position',[20 475 400 300])
title({[choice ] ; ['Height Data']},'Interpreter','none')
```

Matlab Code to Read vk4 file: Display Laser Intensity Image

```
figure(21)

image(myintensity’,'CDataMapping','scaled');
colormap('jet')
colorbar;
daspect([1 1 1]);
set(gcf,'Position',[20 850 400 300])
title({[choice ] ; ['Gamma Corrected Laser Intensity
Data']},'Interpreter','none')
```

Matlab Code to Read vk4 file: Display RGB Laser Optical Images

```
figure(1)

image(reshape(CF(:,1),[colorrows colorcols])',’CDataMapping','scaled')
colormap('jet')
colorbar;
daspect([1 1 1]);
set(gcf,'Position',[20 100 400 300])
title({[choice ] ; ['Blue Laser Optical Data']},'Interpreter','none')

figure(2)

image(reshape(CF(:,2),[colorrows colorcols])',’CDataMapping','scaled');
colormap('jet')
colorbar;
daspect([1 1 1]);
set(gcf,'Position',[425 100 400 300])
title({[choice ] ; ['Green Laser Optical Data']},'Interpreter','none')

figure(3)

image(reshape(CF(:,3),[colorrows colorcols])',’CDataMapping','scaled');
colormap('jet')
colorbar;
daspect([1 1 1]);
set(gcf,'Position',[830 100 400 300])
title({[choice ] ; ['Red Laser Optical Data']},'Interpreter','none')
```

Matlab Code to Read vk4 file: Display RGB Optical Images

```
figure(11)

image(reshape(OF(:,1),[opticalrows opticalcols])',’CDataMapping','scaled')
colormap('jet')
colorbar;
daspect([1 1 1]);
set(gcf,'Position',[1235 100 400 300])
title({[choice ] ; ['Blue Only Optical Data']},'Interpreter','none')

figure(12)

image(reshape(OF(:,2),[opticalrows opticalcols])',’CDataMapping','scaled');
colormap('jet')
colorbar;
daspect([1 1 1]);
set(gcf,'Position',[1640 100 400 300])
title({[choice ] ; ['Green Only Optical Data']},'Interpreter','none')

figure(13)

image(reshape(OF(:,3),[opticalrows opticalcols])',’CDataMapping','scaled');
colormap('jet')
colorbar;
daspect([1 1 1]);
set(gcf,'Position',[2045 100 400 300])
title({[choice ] ; ['Red Only Optical Data']},'Interpreter','none')
```

Matlab Code to Read vk4 file: Display Beginning Data and Save Images to Workspace

```
myheight(1:14,1:4)'
mycolors(1:14,1:4,1)'
mycolors(1:14,1:4,2)'
mycolors(1:14,1:4,3)'
myoptical(1:14,1:4,1)'
myoptical(1:14,1:4,2)'
myoptical(1:14,1:4,3)'
myintensity(1:14,1:4)'
savefilename=replace(myfile,'vk4','mat')
save temp filename myheight mycolors myoptical myintensity heightrows
heightcols colorrows colorcols opticalrows opticalcols intensityrows
intensitycols
movefile('tempfilename.mat',savefilename)
```

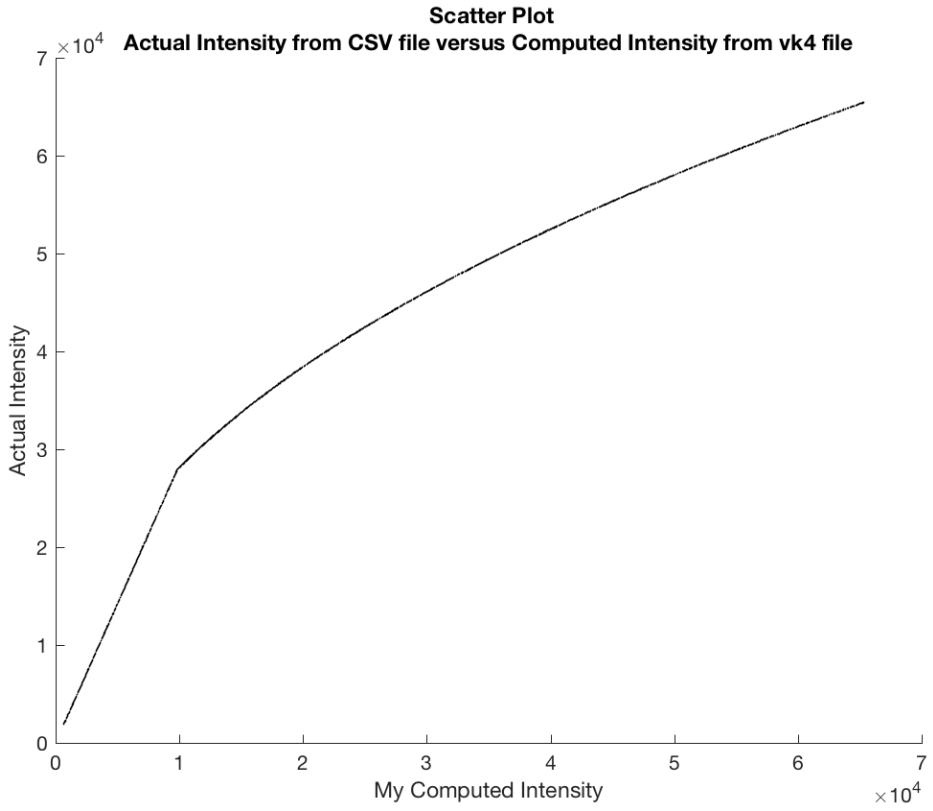
Matlab Code to Read vk4 file: Gamma Correction Subroutine

```
functiony = fmyintensity(x)
% gamma power
mygamma=.45;
% slope
aequal=2.8389;
% intercept
bequal=-.3851;
% slope
aroot=445.7088;
% intercept
broot=.5506;
% change point
linecut=38*256+102;
% maximum range
linemax=256^2-1;
% % % % % % % % linecut=38*256+102
% % % % % % % % linemax=255*256+255
y=zeros(size(x));
y = y + (x>=0&x<=linecut).*(aequal.*x+bequal);
y = y + (x>linecut&x<=linemax).*(aroot.*x.^(mygamma)+broot);
end
```

Reading Intensity from the vk4 file

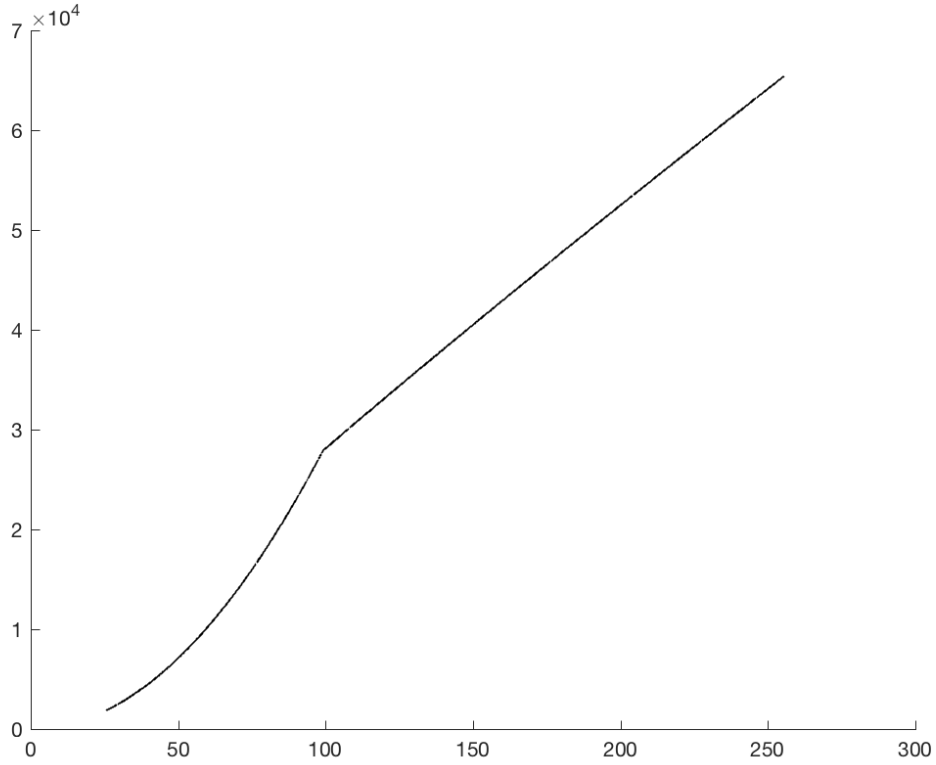
Linear in the low range

- Fbase2=[1 16^2]';
- myintensity=IF*Fbase2;
- myi=IF*Fbase2;



Linear in the high range

- scatter(myi.^.5,test4,sz,'filled','k'`



Gamma Correction for vk4 files

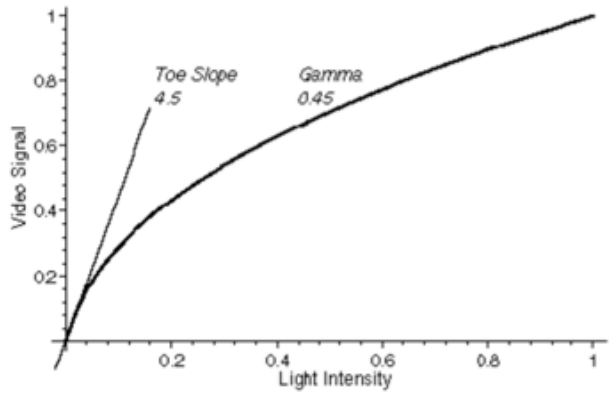
http://archive.is/iuUnG

5. What is *gamma correction*?

In a video system, linear-light intensity is transformed to a nonlinear video signal by *gamma correction*, which is universally done at the camera. The Rec. 709 transfer function [2] takes linear-light intensity (here R) to a nonlinear component (here R'), for example, voltage in a video system:

$$R'_{709} = \begin{cases} 4.5R, & R \le 0.018 \\ 1.099R^{0.45} - 0.099, & 0.018 < R \end{cases}$$

The linear segment near black minimizes the effect of sensor noise in practical cameras and scanners. Here is a graph of the Rec. 709 transfer function, for a signal range from zero to unity:



An idealized monitor inverts the transform:

$$R = \begin{cases} \dfrac{R'_{709}}{4.5}, & R'_{709} \le 0.081 \\ \left(\dfrac{R'_{709} + 0.099}{1.099}\right)^{\frac{1}{0.45}}, & 0.081 < R'_{709} \end{cases}$$

http://archive.is/iuUnG

Real monitors are not as exact as this equation suggests, and have no linear segment, but the precise definition is necessary for accurate intermediate processing in the linear-light domain. In a colour system, an identical transfer function is applied to each of the three *tristimulus* (linear-light) RGB components. See *Frequently Asked Questions about Colour*.

By the way, the nonlinearity of a CRT is a function of the electrostatics of the cathode and the grid of an electron gun; it has nothing to do with the phosphor. Also, the nonlinearity is a power function (which has the form $f(x) = x^{\wedge a}$), not an exponential function (which has the form $f(x) = a^{\wedge x}$). For more detail, read Poynton's article [3].

19. References

[1] Publication CIE No 15.2, *Colorimetry, Second Edition* (1986), Central Bureau of the Commission Internationale de L'Éclairage, Vienna, Austria.

[2] ITU-R Recommendation BT.709, *Basic Parameter Values for the HDTV Standard for the Studio and for International Programme Exchange* (1990), [formerly CCIR Rec. 709], ITU, 1211 Geneva 20, Switzerland.

[3] Charles A. Poynton, "Gamma and Its Disguises" in *Journal of the Society of Motion Picture and Television Engineers*, Vol. **102**, No. 12 (December 1993), 1099-1108, also available on the Internet (PDF format, 138 KB).

[4] Charles A. Poynton, "Gamma on the Apple Macintosh" (PDF format, 108 KB).

Computing the Gamma Intensity Correction

- Find Gamma Correction Function Coefficients for These Files
- Gamma = .45
- Cutoff = 9830 (directly applied to binary data value (BDV))
- First Line $2.8389 \times BDV - .3851$, $BDV \leq 9830$
- Second Line $445.7088 \times BDV^{0.45} + .5506$, $BDV > 9830$
- Result is the Intensity (±1)

Observations

- Last few lines of the intensity file are duplicated in TA_Stamp_20X_S1_Y1_X1.vk4 file
- Intensity file significant digits is exactly the maximum of the RGB channels

Check Read:
Matches values from vk4 file and csv files

- TA_Stamp_20X_S1_Y1_X1.vk4
- HCL_Pol_no_O2_Sec1_All_Y2_X4