

Panasonic



3DLUT

REFERENCE MANUAL

November 28, 2014

Revision history

Revision	Date	Explanation
Rev.1.0	November 28, 2014	First Edition

Table of Contents

1. Introduction
2. Definition of 3DLUT
3. Example of 3DLUT file

1. Introduction

This document describes the 3D lookup table file specifications used in Panasonic's "Varicam" camera.

2. File definition of 3DLUT file

The extension of this 3DLUT file is .vlt and ASCII character encoding is used. The following rules are followed in the content.

Lines 1 to 4 of the file are the file header, and output values are noted in lines 5 and later.

Line 1 defines the 3DLUT version that for version 1.0 being as follows:

```
# panasonic vlt file version 1.0
```

Line 2 shows the reference file the 3DLUT file is based on. For example, referencing file name 1234.vlt would be noted as follows.

```
# source vlt file "1234.vlt"
```

However, even if the reference file does not need to be noted, the following comment must be used without omitting.

```
# source vlt file ""
```

Line 3 shows the number of point for 3D cubes. 17point 3D cubes shown as follows.

```
LUT_3D_SIZE 17
```

Line 4 notes line breaks.

Line 5 and subsequent lines note output values at RGB coordinates in the order R, G, B on one line. Output values are noted in integers from 0 to 4095, and total grid points—number of lines—is 17x17x17=4913. The 17 point sampling intervals of input values are 0 256 512 768 ... 3584 3840 4095. Order of grid points is incremented with R first followed by G and B. In other words, the line 5 is R0, G0, B0, line 6 is R1, G0, B0, incremented subsequently to R16, G16, B16.

Also, character separators between channels are spaces (ASCII 0x20).

3. 3DLUT file example

Part of a 3DLUT file is shown below. Here, only a few lines are shown, and representations are abbreviated by "...".

```
# panasonic vlt file version 1.0
# source vlt file "1234.vlt"
LUT_3D_SIZE 17
```

```
0 0 0
256 0 0
512 0 0
... ..
4095 0 0
0 256 0
256 256 0
512 256 0
... ..
4095 256 0
... ..
4095 4095 0
0 0 256
256 0 256
512 0 256
... ..
4095 4095 4095
```