Standard Specification for

File Format of Two-Dimensional and Three-Dimensional (2D/3D) Pavement Image Data

AASHTO Designation: MP 47-21[1]

First Published: 2021

Technical Subcommittee: 5a, Pavement Measurement

AASHTO

American Association of State Highway and Transportation Officials
555 12th Street NW, Suite 1000
Washington, DC 20004

Standard Specification for

File Format of Two-Dimensional and Three-Dimensional (2D/3D) Pavement Image Data

AASHTO Designation: MP 47-20[1]

AASHTO

First Published: 2021

Technical Subcommittee: 5a, Pavement Measurement

1. SCOPE

1.1. This specification describes the standard file format of two-dimensional (2D) and three-dimensional (3D) pavement image data used to determine pavement surface condition and profiles.

1.2. This specification describes the data elements stored in a binary file.

1.3. This specification is designed to be independent of hardware platform, computer language, and operating system.

1.4. This standard does not purport to address all the safety concerns, if any, associated with its use. It is the responsibility of the user of this standard to establish appropriate safety and health practices and determine the applicability of regulatory limitations prior to use.

2. REFERENCED STANDARDS

2.1. *ASTM Standard*:

- E867, Terminology Relating to Vehicle-Pavement Systems

2.2. *IEEE Standard*:

- IEEE 754, IEEE Standard for Floating-Point Arithmetic

3. TERMINOLOGY

3.1. *Definitions*:

3.1.1. Terminology used in this specification conforms to the definitions included in ASTM E867.

3.2. *Definitions of Terms Specific to This Standard*:

3.2.1. *array:*

3.2.1.1. *array (numeric data type)*—Sequence of data of the specified numeric data type. The size of the array will be stored separately.

3.2.1.2. *array (string)*—ASCII strings separated by null characters. A null character follows the last string.

3.2.2. *bit depth*—Number of binary digits (bits) used to represent a value of image intensity or range.

3.2.3. *compatibility:*

3.2.3.1. *backward compatibility*—Ability of a software system, such as a pavement image viewer, to read an earlier version of the standard file format.

3.2.3.2. *forward compatibility*—Ability of a software system, such as a pavement image viewer, to read a future version of the standard file format.

3.2.4. *compression*—Method to reduce image size by encoding information into fewer bits than the original representation.

3.2.4.1. *lossless compression*—Compression method in which no information is lost.

3.2.4.2. *lossy compression*—Compression method in which information may be lost.

3.2.5. *compression quality*—Measurement of how much information is lost during compression. In this specification, compression quality is measured using peak signal-to-noise ratio (PSNR), the ratio of the maximum possible power of a signal to the power of noise that corrupts the signal fidelity. When lossless compression is used, the PSNR is infinite.

The equation for PSNR is:

$$\mathrm{PSNR} = 10 \times \log_{10}\left(\frac{MAX_I^2}{MSE}\right) \quad (1)$$

where:

$$MSE = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\left(Iij - Tij\right)^2 \quad (2)$$

I represents the reference image and T represents the image to be tested. The dimensions of both images are $M \times N$. The MSE is the mean square error between the test image and the reference image. MAX_I is the maximum possible value of a pixel. For an 8-bit intensity image, it is $2^8 - 1 = 255$, and for a 16-bit range image, it is $2^{16} - 1 = 65535$.

3.2.6. *FIPS state codes*—Federal Information Processing Standard state code. FIPS state codes were numeric and two-letter alphabetic codes defined in U.S. Federal Information Processing Standard Publication.

3.2.7. *floating point number:*

3.2.7.1. *Float32*—Data type for a 32-bit binary floating point number, defined as “binary 32” by IEEE 754.

3.2.7.2. *Float64*—Data type for a 64-bit binary floating point number, defined as “binary 64” by IEEE 754.

3.2.8. *global positioning system (GPS)*—Network of navigation satellites used to determine the location of receiving devices.

3.2.9. *integer:*

3.2.9.1. *signed*—Integer capable of representing negative and positive values.

3.2.9.1.1. *Int32*—Data type for a 32-bit, signed integer.