Natural Vision multispectral image metadata format, XML schema (NVXML), Specification

**Version 1.20 2012/09/03**

# 1. Introduction

## 1.1 Scope of this specification

This document specifies the method to describe the metadata for a multispectral image using XML (Extensible Markup Language), recommended by W3C (World Wide Web Consortium). This specification is based on the XML format developed by Natural Vision (NV) project of NICT (formerly TAO).

The scope of this specification is to define the data structure for the metadata for multispectral images that manages the color information. The class of multispectral images that include the visible range of spectra is employed for both the color reproduction and the spectral image analysis. It is assumed that the metadata is attached to the multispectral image data. As the metadata is described by XML, it can be enclosed in different image data formats, where the metadata format is independent of image file format.

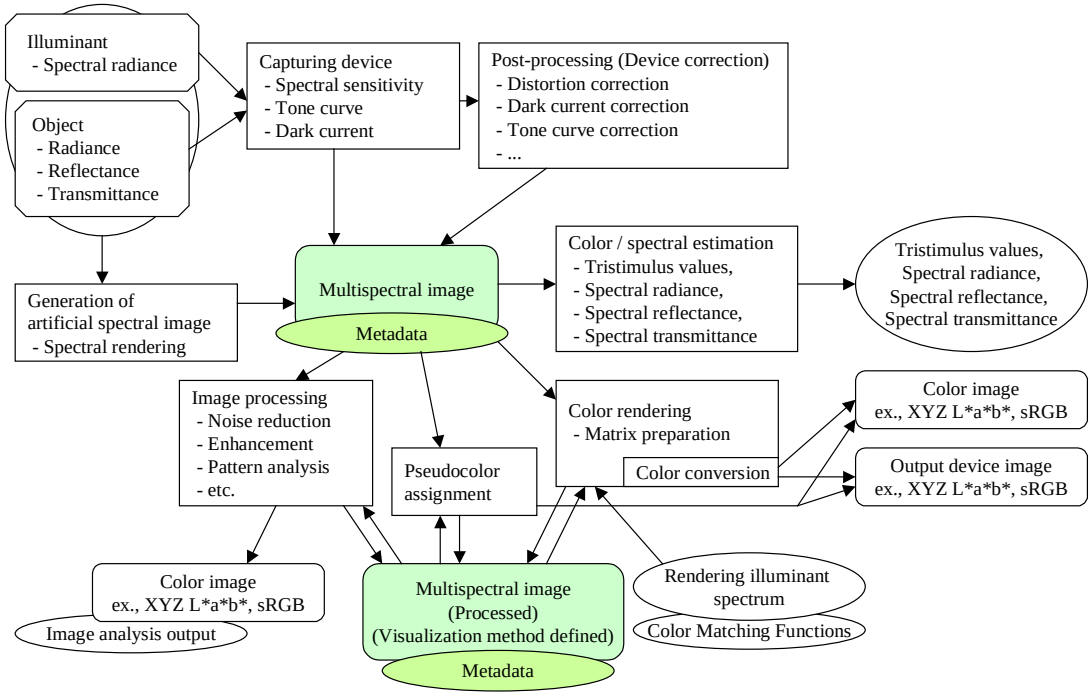


Figure 1. Overview of the workflow of multispectral imaging systems

## 1.2 Overview of the metadata format

The metadata is constituted by a root object (Nvision), as well as three node objects "NvisionImage", "NvisionInput" and "NvisionConversion", as shown in figure 2.

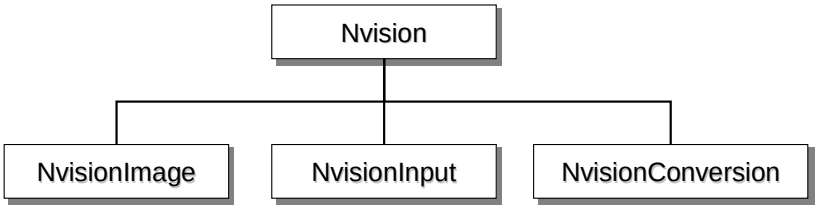


Figure 2. Overview of the matadata structure

## 2. The root object "Nvision"

The elements of the root object "Nvision" are summarized in the table 1.

Table 1 Root object "Nvision"

| Item | Name | Data type | Description | Requirement level |
|---|---|---|---|---|
| Root Object | Nvision | NvisionType | The name of the root object | REQUIRED |
| Element (Object) | NvisionImage | NvisionImageType | [See Section 3] | REQUIRED |
| | NvisionInput | NvisionInputType | [See Section 4] | REQUIRED |
| | NvisionConversion | NvisionConvType | [See Section 5] | OPTIONAL* |

* If the requirement level of an object (or an element) is "optional", the object (or element) may not be recorded. The requirement level of the element contained in the optionally required object (or element) is valid only if the containing object (or element) is present.

### 2.1 The XML schema definition of "Nvision"

The NVXML schema specification begins with the XML declaration including the version number of XML schema and the character encoding used in it. In this schema the target namespace should be declared.
The root object "Nvision" is defined by the XML schema as follows;

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema targetNamespace ="http://nvision.jp/NvXmlSchema"
           xmlns:xs="http://www.w3.org/2001/XMLSchema"
           xmlns:msdata="urn:schemas-microsoft-com:xml-msdata"
           xmlns:d2p1="http://nvision.jp/NvXmlSchema"
           >

  <xs:element name="Nvision" type="d2p1:NvisionType" msdata:IsDataSet="true"/>
  <xs:complexType name="NvisionType">
    <xs:sequence>
      <xs:element name="NvisionImage" type="d2p1:NvisionImageType" />
      <xs:element name="NvisionInput" type="d2p1:NvisionInputType" />
      <xs:element name="NvisionConversion" type="d2p1:NvisionConvType" minOccurs="0" />
    </xs:sequence>
  </xs:complexType>

  <!-- The definitions of the elements [See later sections] shall be described  -->.

</xs:schema>
```

### 2.2 Example of the "Nvision"

```
<?xml version="1.0" encoding="utf-8" ?>
<d2p1:Nvision xmlns:d2p1=http://nvision.jp/NvXmlSchema
              xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
              xsi:schemaLocation="http:// nvision.jp /NvXmlSchema nvxml.xsd">

  …

</d2p1:Nvision>
```

3. "NvisionImage" object

"NvisionImage" object shall contain two elements, "ImageCreateInfo" and "ImageInfo" as described in subsections 3.1 and 3.2.

Table 2 "NvisionImage" object

| Element name | Data type | Description | Requirement level |
|---|---|---|---|
| ImageCreateInfo | ImageCreateInfoType | Information related to the image data creation [See Subsection 3.1] | REQUIRED |
| ImageInfo | ImageInfoType | Information related to the image data structure [See Subsection 3.2] | REQUIRED |

### 3.1 "ImageCreateInfo" element

"ImageCreateInfo" element contains the information related to the image data creation.

Table 3 " ImageCreateInfo" element

| Element name | Data type | Description | Requirement level |
|---|---|---|---|
| Signature | string | Signature of this metadata. Fixed to ASCII string "NVXML" | REQUIRED |
| Version | string | Version of NVXML (Current version is 1.20) | REQUIRED |
| Creator | string | Human readable name of the creator of this NVXML data | OPTIONAL |
| CreationDate | string | Created date of this NVXML data in YYYY-MM-DD format defined in ISO 8601 (http://www.w3.org/TR/NOTE-datetime) | OPTIONAL |
| LastUpdate | string | Date of the last update in YYYY-MM-DD format defined in ISO 8601 (http://www.w3.org/TR/NOTE-datetime) | OPTIONAL |
| Rights | string | Owner of this NVXML | OPTIONAL |
| Comment | string | Comments | OPTIONAL |

### 3.2 "ImageInfo" element

"ImageInfo" element contains the information related to the image data structure.

Table 4 "ImageInfo" element

| Element name | Data type | Description | Requirement level |
|---|---|---|---|
| ImageType | ImageTypeType | The type of image data, one of the followings;<br>SOURCE: The image data is the raw output from the image capturing or input device (or the raw computer generated image).<br>PROCESSED: Processed data, including that at least one of the input device characteristics, that is, the bias level, the exposure time of each channel, the iris, or the tone curve is corrected. In this case, the image data may hold the device dependent data.<br>COLOR: The image data represent the values in the color space specified in "ColorSpace" element. In this case, "ImageBands" element = 3. | REQUIRED |

| | | | |
|---|---|---|---|
| ColorSpace | ColorSpaceType | The color space of the image data, which is specified when "ImageType" element = "COLOR".<br>The data should be one of the followings;<br>XYZ: the image data of the CIE 1931XYZ specified by ref. [7].<br>LAB: the image data of the CIE 1976 L*a*b* specified by ref. [7].<br>sRGB: the image data of the RGB specified by ref. [8].<br>xvYCC: the image data of the YCC specified by ref. [9]. | OPTIONAL |
| ImageBands | unsignedInt | The number of channels | OPTIONAL |
| BitSizePerBand | unsignedInt | Bit size of each pixel value (Same value for all bands) | OPTIONAL |
| DataType | DataTypeType | Data type of the image, one of the followings;<br>UINTxx: xx-bits unsigned integer<br>(UINT8, UINT16, UINT32, UINT64)<br>INTxx: xx-bits signed integer<br>(INT8, INT16, INT32, INT64)<br>U8FIXED8: 16-bit unsigned fixed-point number with 8-bit fractional part.<br>U16FIXED16: 32-bit unsigned fixed-point number with 16-bit fractional part.<br>S7FIXED8: 16-bit signed fixed-point number with 8-bit fractional part.<br>S15FIXED16: 32-bit signed fixed-point number with 16-bit fractional part.<br>UFLOAT: 32-bit unsigned floating point number<br>FLOAT: 32-bit signed floating point number | OPTIONAL |
| ImageWidth | unsignedInt | The pixel number of the image width | OPTIONAL |
| ImageHeight | integer | The pixel number of the image height (If ImageHeight>0, the origin is the bottom-left of the image. If ImageHeight<0, the origin is the top-left of the image.) | OPTIONAL |
| DataOrder | DataOrderType | The order of binary data in the image, one of the followings;<br>"BSQ": Band sequential<br>"BIL": Band interleaved by line<br>"BIP": Band interleaved by pixel | OPTIONAL |

### 3.3 The XML schema definition of "NvisionImage" object

The "NvisionImage" object is defined by the XML schema as follows;

```
<xs:complexType name="NvisionImageType">
  <xs:sequence>
    <xs:element name="ImageCreateInfo" type="d2p1:ImageCreateInfoType"/>
    <xs:element name="ImageInfo" type="d2p1:ImageInfoType"/>
  </xs:sequence>
</xs:complexType>
```

```
<xs:complexType name="ImageCreateInfoType">
  <xs:sequence>
    <xs:element name="Signature" type="xs:string"/>
    <xs:element name="Version" type="xs:string"/>
    <xs:element name="Creator" type="xs:string" minOccurs="0"/>
    <xs:element name="CreationDate" type="xs:string" minOccurs="0"/>
    <xs:element name="LastUpdate" type="xs:string" minOccurs="0"/>
    <xs:element name="Rights" type="xs:string" minOccurs="0"/>
    <xs:element name="Comment" type="xs:string" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

```
<xs:complexType name="ImageInfoType">
  <xs:sequence>
    <xs:element name="ImageType" type="d2p1:ImageTypeType"/>
    <xs:element name="ColorSpace" type="d2p1:ColorSpaceType" minOccurs="0"/>
    <xs:element name="ImageBands" type="xs:unsignedInt" minOccurs="0"/>
    <xs:element name="BitSizePerBand" type="xs:unsignedInt" minOccurs="0"/>
    <xs:element name="DataType" type="d2p1:DataTypeType" minOccurs="0"/>
    <xs:element name="ImageWidth" type="xs:unsignedInt" minOccurs="0"/>
    <xs:element name="ImageHeight" type="xs:integer" minOccurs="0"/>
    <xs:element name="DataOrder" type="d2p1:DataOrderType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

```
<xs:simpleType name="ImageTypeType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="SOURCE"/>
    <xs:enumeration value="PROCESSED"/>
    <xs:enumeration value="COLOR"/>
  </xs:restriction>
</xs:simpleType>
```

```
<xs:simpleType name="ColorSpaceType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="XYZ"/>
    <xs:enumeration value="LAB"/>
    <xs:enumeration value="sRGB"/>
    <xs:enumeration value="xvYCC"/>
  </xs:restriction>
</xs:simpleType>
```

```
<xs:simpleType name="DataTypeType">
  <xs:restriction base="xs:string">
```

```
      <xs:pattern value ="UINT+(64|32|16|8)"/>
      <xs:pattern value ="INT+(64|32|16|8)"/>
      <xs:pattern value ="(U8FIXED8|U16FIXED16|S7FIXED8|S15FIXED16|UFLOAT|FLOAT)"/>
    </xs:restriction>
  </xs:simpleType>
```

```
  <xs:simpleType name="DataOrderType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="BSQ"/>
      <xs:enumeration value="BIL"/>
      <xs:enumeration value="BIP"/>
    </xs:restriction>
  </xs:simpleType>
```

### 3.4 Example of "NvisionImage"

```
<NvisionImage>
  <ImageCreateInfo>
        <Signature>NVXML</Signature>
        <Version>1.20</Version>
        <Creator> T.Yamada</Creator>
        <CreationDate>20120822</CreationDate>
        <LastUpdate>20120903</LastUpdate>
        <Rights>XYZ Co.</Rights>
        <Comment>Natural Vision Project</Comment>
  </ImageCreateInfo>
  <ImageInfo>
        <ImageType>SOURCE</ImageType>
        <ImageBands>16</ImageBands>
         <BitSizePerBand>16</BitSizePerBand>
         <DataType>UINT16</DataType>
          <ImageWidth>1280</ImageWidth>
         <ImageHeight>-1024</ImageHeight>
        <DataOrder>BIP</DataOrder>
  </ImageInfo>
</NvisionImage>
```

## 4. "NvisionInput" object

"NvisionInput" object may contain five elements, "InputDevInfo", "InputDevData", "InputImageInfo", "InputIllu" and "SubjectSpecMatrix" as described in subsections 4.1 to 4.5. In the case that the image is given by a non-device color space, such as XYZ, this object can be omitted. The created date of this object data may be described in the "InputDate" attribute.

Table 5 "NvisionInput" object

| Element name | Data type | Description | Requirement level |
|---|---|---|---|
| InputDevInfo | InputDevInfoType | Information for the identification of the input device [See Subsection 4.1] | OPTIONAL |
| InputDevData | InputDevDataType | Information for the input device characterization [See Subsection 4.2] | OPTIONAL |
| InputImageInfo | InputImageInfoType | Specifications of each band of the image [See Subsection 4.3] | OPTIONAL |
| InputIllu | InputIlluType | The spectrum of the input illuminant [See Subsection 4.4] | OPTIONAL |
| SubjectSpecMatrix | SubjectSpecMatrixType | Statistical data of the object spectral characteristics [See Subsection 4.5] | OPTIONAL |

### 4.1 "InputDevInfo" element

"InputDevInfo" element may contain the information related to the identification of the input device.

Table 6 "InputDevInfo" element

| Element name | Data type | Description | Requirement level |
|---|---|---|---|
| InputDevName | string | The name of the input device. | OPTIONAL |
| InputDevDescription | string | Description of the input device | OPTIONAL |
| InputDevManufacturer | string | Manufacturer name of the input device | OPTIONAL |

### 4.2 " InputDevData " element

"InputDevData" element may contain the information related to the characterization of the input device, such as the spectral sensitivities, dark current levels, tone curves, and noise levels. The details of these elements are described below;

Table 7 "InputDevData" element

| Element name | Data type | Description | Requirement level |
|---|---|---|---|
| SpecSensiData | SpecSensiDataType | Spectral sensitivity of each band [See description] | OPTIONAL |
| CoeffData1 | CoeffData1Type | First coefficient for the spectral sensitivity of each band [See description] | OPTIONAL |
| CoeffData2 | CoeffData2Type | Second coefficient for the spectral sensitivity of each band [See description] | OPTIONAL |
| CoeffData3 | CoeffData3Type | Third coefficient for the spectral sensitivity of each band [See description] | OPTIONAL |
| DarkCurrentData | DarkCurrentDataType | Dark current data of each band [See description] | OPTIONAL |
| NoiseData | NoiseDataType | Noise level of each band [See description] | OPTIONAL |

| | | | |
|---|---|---|---|
| ToneCurvesData | ToneCurvesDataType | Tone Curve Data (A set of sampled pixel values and the corresponding signals linear to the light intensity of each band) [See description] | OPTIONAL |

It is assumed that the pixel values of a M-band multispectral image, dj (j=1,...,M), can be modeled independently from the pixel location on the image as

$$d_j = t_j\left[ clip\left[ \int_\lambda h'_j(\lambda)s(\lambda)d\lambda + b_j + n_j \right] \right] \quad \text{(j=1, …,M),} \qquad (1)$$

$$h'_j(\lambda) = e_j f_j c_j h_j(\lambda) \quad ,$$

where

$s(\lambda)$ : The spectral radiance of an object to be imaged by the input device

$h_j(\lambda)$ : The spectral sensitivity of j-th band of the input device under a certain measurement condition

$c_j$ : The first level correction value of the spectral sensitivity of j-th band of the input device

$f_j$ : The second level correction value of the spectral sensitivity of j-th band of the input device

$e_j$ : The third level correction value of the spectral sensitivity of j-th band of the input device

$b_j$ : The fixed bias value additive to the j-th band signal linear to the light intensity

$n_j$ : The variable bias value additive to j-th band signal linear to the light intensity

$clip[in]$: The operator to map the input value "*in*" exceeding 1.0 to 1.0

$t_j[in]$: The operator to transform the input value "*in*" non-linearly. In this operation the range is also changed so that the maximum value of the "*in*" ( i.e.,1.0) is mapped to N.

N: The maximum value of the pixel value (N should be $2^{\text{BitSizePerBand}}-1$ when BitSizePerBand in "ImageInfo" element is described)

If the object is reflective or transparent, the spectral radiance of the object $s(\lambda)$ [W/m$^2$/sr] is assumed to satisfy the equation (2),

$$s(\lambda) = l(\lambda)r(\lambda), \qquad (2)$$

where $l(\lambda)$ represents the input illuminant spectrum and $r(\lambda)$ (0-1) represents the spectral reflectance or transmittance of the object.

If the spectral sensitivities of the input device and the input illuminant spectrum can not be obtained separately, as in the case that the image data is input by an image scanner, the total spectral characteristics which includes the spectral sensitivities of the input device and the input illuminant spectrum are described as $h_j(\lambda)$ and 1.0 is described as the value of the input illuminant spectrum $l(\lambda)$ of each wavelength.

If the different band image data are input under different input illuminant spectrum, the total spectral characteristics which include the spectral sensitivities of the input device and the input illuminant spectrum are described as $h_j(\lambda)$ and 1.0 is described as the value of the input illuminant spectrum $l(\lambda)$ of each wavelength.

If the illuminance of the input illuminant varies depending on the object location, the input illuminant spectrum $l(\lambda)$ remained to be described as location-independent data and the definition of the spectral reflectance or transmittance of the object $r(\lambda)$ is modified to satisfy the equation (2) in the geometrical condition that the image data is input.

Each element of "InputDevData", which is related to the equation (1), is defined as follows.

#### 4.2.1 SpecSensiData

The spectral sensitivity values $h_j(\lambda_i)$ at sampling wavelengths $\lambda_i=\lambda_0+(i-1)\Delta$ (i=1,…,R; j=1,…,M) are described in the "SpceSensiValue" (ArrayOfFloatType) element,
where

$\Delta$ : the wavelength interval in nm,
$\lambda_0$ : the shortest wavelength in nm,
R : the number of sampling wavelengths.

The R and the M are described in the "Row" and the "Column" element of the "MatrixAttrGroup" attribute respectively (Refer to section 6 “Definition of basic data type” concerning the description order of the data corresponding to each attribute).

The $\lambda_0$, the R (same R value described in the "Row" element of the "MatrixAttrGroup" attribute as mentioned above) and the $\Delta$ are described in the "ShortWaveLength", the "DataNumber" and the "WaveInterval" element of the "SpecDataAttrGroup" attribute respectively.

If the spectral sensitivity is provided as a set of absolute values satisfying the equation (1), "Absolute" is described in the "DEF" element of the "DataDefinitionAttrGroup" attribute. If the spectral sensitivity is provided as a set of relative values, "Relative" is described in the "DEF" element of the "DataDefinitionAttrGroup" attribute.

#### 4.2.2 CoeffData1, CoeffData2 and CoeffData3

The level correction values $c_j$, $f_j$ and $e_j$ (j=1,…,M) are described in the "CoeffValue1"(ArrayOfFloatType), the "CoeffValue2"(ArrayOfFloatType) and the "CoeffValue3"(ArrayOfFloatType) element respectively. These level correction values are intended to correct the levels of the spectral sensitivities to satisfy the equation (1). These corrections are useful in the situation that the settings of the image capturing device such as the exposure time and the aperture value at the image capturing are different from those set at the measurement of the spectral sensitivities $h_j(\lambda)$. In each coefficient element the band number M is described in the "VectorDim" element of the "VectorAttrGroup" attribute. If these "CoeffData" elements are not described, the level correction values of all the bands are regarded to be 1.0.

#### 4.2.3 DarkCurrentData

The dark current data $b_j$ (j=1,…,M) are described in the "DarkCurrentValue"(ArrayOfFloatType) element. The band number M is described in the "VectorDim" element of the "VectorAttrGroup" attribute. If the "DarkCurrentData" element is not described, the values of the all bands are regarded to be zero.

#### 4.2.4 NoiseData

The statistical values $\left\langle \left| n_j \right|^2 \right\rangle$ ($\langle\ \rangle$ represents statistical expectation) of noise $n_j$ (j=1,…,M) are described in the "NoiseValue"(ArrayOfFloatType) element. The band number M is described in the "VectorDim" element of the "VectorAttrGroup" attribuete. If the "NoiseData" element is not described, the values of all the bands are regarded to be zero.

#### 4.2.5 ToneCurvesData

The sampled pixel values $d_k$ (k=1,…,L) and the corresponding values linear to the light intensity of each band $in_{jk}$ ($d_k = t_j\lfloor in_{jk} \rfloor$) (k=1,…,L; j=1,…,M) are described in the "CurveValue"(ArrayOfFloatType) element in this order. The maximum values of $in_{jk}$ and $d_k$ should be 1.0 and N respectively. The number of sampled pixel values L and the data set number 1+M are described in the "Row" and the "Column" element of the "MatrixAttrGroup" attribute respectively. If the "ToneCurvesData" element is not described, the relationships between $in_{jk}$ and $d_k$ are regarded to be linear.

### 4.3 "InputImageInfo" element

"InputImageInfo" element may contain the information related to the specifications of each band of the image.

Table 8 "InputImageInfo" element

| Element name | Data type | Description | Requirement level |
|---|---|---|---|
| BandName | BandNameType | The name of each band | OPTIONAL |
| IrisSetting | IrisSettingType | The iris setting value of each band | OPTIONAL |
| ExposureTimeSetting | ExposureTimeSettingType | The exposure time setting value of each band | OPTIONAL |

#### 4.3.1 BandName

The name of each band is described as space-separated string in the "BandNameData" element (string). The number of bands is described in the "VectorDim" element of the "VectorAttrGroup" attribute.

#### 4.3.2 IrisSetting

The iris setting of each band is described as space-separated string Fxx (xx: F number) in the "IrisSettingData" element (string). The number of bands is described in the "VectorDim" element of the "VectorAttrGroup" attribute.

#### 4.3.3 ExposureTimeSetting

The exposure time setting value on the second time scale of each band is described as space-separated string in the "ExposureTimeSettingData" element (string). The number of bands is described in the "VectorDim" element of the "VectorAttrGroup" attribute.

### 4.4 "InputIllu" element

"InputIllu" element may contain the information related to the input illuminant spectrum.

The sampled discrete data $l(\lambda_i)$ (i=1,…,R) of the input illuminant spectrum $l(\lambda)$ at sampling wavelengths $\lambda_i=\lambda_0+(i-1)\Delta$ (i=1,…,R) are described in the "InputSpceData"(ArrayOfFloatType) element,
where

$\Delta$ : the wavelength interval in nm,
$\lambda_0$ : the shortest wavelength in nm,
R : the number of sampling wavelengths.

The number of sampling wavelengths R is described in the "VectorDim" element of the "VectorAttrGroup" attribute.

The $\lambda_0$ , the R (same R value described in the "VectorDim" element of the "VectorAttrGroup" attribute as mentioned above) and the $\Delta$ are described in the "ShortWaveLength", the "DataNumber" and the "WaveInterval" element of the "SpecDataAttrGroup" attribute respectively.

## 4.5 “SubjectSpecMatrix” element

"SubjectSpecMatrix" element may contain the statistical data of the object spectral characteristics.

Table 9 "SubjectSpecMatrix" element

| Element name | Data type | Description | Requirement level |
|---|---|---|---|
| EigenRefData | EigenRefDataType | The principal components of the spectral reflectance or transmittance of the object | OPTIONAL |
| EigenSpecData | EigenSpecDataType | The principal components of the spectral radiance of the object | OPTIONAL |

### 4.5.1 EigenRefData

The P eigen vectors $v_p(\lambda_i)$ (p=1,…,P; i=1,…,R) and the corresponding eigen values $w_p$ (p=1,…,P) of the spectral reflectance or transmittance of objects at sampling wavelengths $\lambda_i=\lambda_0+(i-1)\Delta$ (i=1,…,R) are described in the "EigenRefValue"(ArrayOfFloatType) element,

where

$\Delta$ : the wavelength interval in nm,

$\lambda_0$ : the shortest wavelength in nm,

R : the number of sampling wavelengths.

The data number R+1 and the data set number P are described in the "Row" and the "Column" element of the "MatrixAttrGroup" attribute respectively. The description order of these data is as follows; $v_1(\lambda_1)\ \ v_2(\lambda_1)\ \ \ldots\ \ v_P(\lambda_1)\ \ v_1(\lambda_2)\ \ \ldots\ \ v_P(\lambda_R)\ \ w_1\ \ \ldots\ \ w_P$.

The $\lambda_0$, the R (same R value described in the "Row" element of the "MatrixAttrGroup" attribute as mentioned above) and the $\Delta$ are described in the "ShortWaveLength", the "DataNumber" and the "WaveInterval" element of the "SpecDataAttrGroup" attribute respectively.

### 4.5.2 EigenSpecData

The P eigen vectors $v_p(\lambda_i)$ (p=1,…,P; i=1,…,R) and the corresponding eigen values $w_p$ (p=1,…,P) of the spectral radiance of objects at sampling wavelengths $\lambda_i=\lambda_0+(i-1)\Delta$ (i=1,…,R) are described in the "EigenSpecValue"(ArrayOfFloatType) element,

where

$\Delta$ : the wavelength interval in nm,

$\lambda_0$ : the shortest wavelength in nm,

R : the number of sampling wavelengths.

The data number R+1 and the data set number P are described in the "Row" and the "Column" element of the "MatrixAttrGroup" attribute respectively. The description order of these data is as follows; $v_1(\lambda_1)\ \ v_2(\lambda_1)\ \ \ldots\ \ v_P(\lambda_1)\ \ v_1(\lambda_2)\ \ \ldots\ \ v_P(\lambda_R)\ \ w_1\ \ \ldots\ \ w_P$.

The $\lambda_0$, the R (same R value described in the "Row" element of the "MatrixAttrGroup" attribute as mentioned above) and the $\Delta$ are described in the "ShortWaveLength", the "DataNumber" and the "WaveInterval" element of the "SpecDataAttrGroup" attribute respectively.

## 4.6 The XML schema definition of "NvisionInput" object

The "NvisionInput" object is defined by the XML schema as follows;

```
<xs:complexType name="NvisionInputType">
    <xs:sequence>
      <xs:element name="InputDevInfo" type="d2p1:InputDevInfoType" minOccurs="0"/>
      <xs:element name="InputDevData" type="d2p1:InputDevDataType" minOccurs="0"/>
      <xs:element name="InputImageInfo" type="d2p1:InputImageInfoType" minOccurs="0"/>
      <xs:element name="InputIllu" type="d2p1:InputIlluType" minOccurs="0"/>
      <xs:element name="SubjectSpecMatrix" type="d2p1:SubjectSpecMatrixType" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute name="InputDate" type="xs:dateTime"/>
  </xs:complexType>
```

```
  <xs:complexType name="InputDevInfoType">
    <xs:sequence>
      <xs:element name="InputDevName" type="xs:string" minOccurs="0"/>
      <xs:element name="InputDevDescription" type="xs:string" minOccurs="0"/>
      <xs:element name="InputDevManufacturer" type="xs:string" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
```

```
  <xs:complexType name="InputDevDataType">
    <xs:sequence>
      <xs:element name="SpecSensiData" type="d2p1:SpecSensiDataType" minOccurs="0"/>
      <xs:element name="CoeffData1" type="d2p1:CoeffData1Type" minOccurs="0"/>
      <xs:element name="CoeffData2" type="d2p1:CoeffData2Type" minOccurs="0"/>
      <xs:element name="CoeffData3" type="d2p1:CoeffData3Type" minOccurs="0"/>
      <xs:element name="DarkCurrentData" type="d2p1:DarkCurrentDataType" minOccurs="0"/>
      <xs:element name="NoiseData" type="d2p1:NoiseDataType" minOccurs="0"/>
      <xs:element name="ToneCurvesData" type="d2p1:ToneCurvesDataType" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
```

```
  <xs:complexType name="SpecSensiDataType">
    <xs:sequence>
      <xs:element name="SpecSensiValue" type="d2p1:ArrayOfFloat"/>
    </xs:sequence>
    <xs:attributeGroup ref="d2p1:MatrixAttrGroup"/>
    <xs:attributeGroup ref="d2p1:SpecDataAttrGroup"/>
    <xs:attributeGroup ref="d2p1:DataDefinitionAttrGroup"/>
  </xs:complexType>
```

```
  <xs:complexType name="CoeffData1Type">
    <xs:sequence>
      <xs:element name="CoeffValue1" type="d2p1:ArrayOfFloat"/>
    </xs:sequence>
    <xs:attributeGroup ref="d2p1:VectorAttrGroup"/>
  </xs:complexType>
```

```
  <xs:complexType name="CoeffData2Type">
    <xs:sequence>
      <xs:element name="CoeffValue2" type="d2p1:ArrayOfFloat"/>
    </xs:sequence>
```

```
    <xs:attributeGroup ref="d2p1:VectorAttrGroup"/>
  </xs:complexType>

  <xs:complexType name="CoeffData3Type">
    <xs:sequence>
      <xs:element name="CoeffValue3" type="d2p1:ArrayOfFloat"/>
    </xs:sequence>
    <xs:attributeGroup ref="d2p1:VectorAttrGroup"/>
  </xs:complexType>

  <xs:complexType name="DarkCurrentDataType">
    <xs:sequence>
      <xs:element name="DarkCurrentValue" type="d2p1:ArrayOfFloat"/>
    </xs:sequence>
    <xs:attributeGroup ref="d2p1:VectorAttrGroup"/>
  </xs:complexType>

  <xs:complexType name="NoiseDataType">
    <xs:sequence>
      <xs:element name="NoiseValue" type="d2p1:ArrayOfFloat"/>
    </xs:sequence>
    <xs:attributeGroup ref="d2p1:VectorAttrGroup"/>
  </xs:complexType>

  <xs:complexType name="ToneCurvesDataType">
    <xs:sequence>
      <xs:element name="CurveValue" type="d2p1:ArrayOfFloat"/>
    </xs:sequence>
    <xs:attributeGroup ref="d2p1:MatrixAttrGroup"/>
  </xs:complexType>

  <xs:complexType name="InputImageInfoType">
    <xs:sequence>
      <xs:element name="BandName" type="d2p1:BandNameType" minOccurs="0"/>
      <xs:element name="IrisSetting" type="d2p1:IrisSettingType" minOccurs="0"/>
      <xs:element name="ExposureTimeSettng" type="d2p1:ExposureTimeSettngType" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="BandNameType">
    <xs:sequence>
      <xs:element name="BandNameData" type="xs:string"/>
    </xs:sequence>
    <xs:attributeGroup ref="d2p1:VectorAttrGroup"/>
  </xs:complexType>

  <xs:complexType name="IrisSettingType">
    <xs:sequence>
      <xs:element name="IrisSettingData" type="xs:string"/>
    </xs:sequence>
    <xs:attributeGroup ref="d2p1:VectorAttrGroup"/>
  </xs:complexType>
```

```
<xs:complexType name="ExposureTimeSettingType">
  <xs:sequence>
    <xs:element name="ExposureTimeSettingData" type="xs:string"/>
  </xs:sequence>
  <xs:attributeGroup ref="d2p1:VectorAttrGroup"/>
</xs:complexType>
```

```
<xs:complexType name="InputIlluType">
  <xs:sequence>
    <xs:element name="InputSpecData" type="d2p1:ArrayOfFloat"/>
  </xs:sequence>
  <xs:attributeGroup ref="d2p1:VectorAttrGroup"/>
  <xs:attributeGroup ref="d2p1:SpecDataAttrGroup"/>
</xs:complexType>
```

```
<xs:complexType name="SubjectSpecMatrixType">
  <xs:sequence>
    <xs:element name="EigenRefData" type="d2p1:EigenRefDataType" minOccurs="0"/>
    <xs:element name="EigenSpecData" type="d2p1:EigenSpecDataType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

```
<xs:complexType name="EigenRefDataType">
  <xs:sequence>
    <xs:element name="EigenRefValue" type="d2p1:ArrayOfFloat "/>
  </xs:sequence>
  <xs:attributeGroup ref="d2p1:MatrixAttrGroup"/>
  <xs:attributeGroup ref="d2p1:SpecDataAttrGroup"/>
</xs:complexType>
```

```
<xs:complexType name="EigenSpecDataType">
  <xs:sequence>
    <xs:element name="EigenSpecValue" type="d2p1:ArrayOfFloat"/>
  </xs:sequence>
  <xs:attributeGroup ref="d2p1:MatrixAttrGroup"/>
  <xs:attributeGroup ref="d2p1:SpecDataAttrGroup"/>
</xs:complexType>
```

## 4.7 Example of “NvisionInput”

```
<NvisionInput InputDate="2010-12-31T00:00:00.0000000+09:00">

<InputDevInfo>
  <InputDevName>MSC1000</InputDevName>
  <InputDevDescription>Multispectral Camera</InputDevDescription>
  <InputDevManufacturer> XYZ Co.</InputDevManufacturer>
 </InputDevInfo>

<InputDevData>
  <SpecSensiData Row="401" Column="16" ShortWaveLength="380.0" DataNumber="401"
                 WaveInterval="1.0" DEF="Absolute" >
   <SpecSensiValue CountOfArray="6416">
     <item>0.000000</item>
     <item>0.110000</item>
     <item>0.250000</item>
     …
     <item>0.130000</item>
   </SpecSensiValue>
  </SpecSensiData>
  <CoeffData1 VectorDim="16">
   <CoeffValue1 CountOfArray="16">
     <item>1.000000</item>
     <item>1.100000</item>
     …
     <item>0.900000</item>
   </CoeffValue1>
  </CoeffData1>
  <CoeffData2 VectorDim="16">
   <CoeffValue2 CountOfArray="16">
     <item>0.800000</item>
     <item>1.000000</item>
     …
     <item>1.100000</item>
   </CoeffValue2>
  </CoeffData2>
  <CoeffData3 VectorDim="16">
   <CoeffValue3 CountOfArray="16">
     <item>1.300000</item>
     <item>1.200000</item>
     …
     <item>0.800000</item>
   </CoeffValue3>
  </CoeffData3>
  <DarkCurrentData VectorDim="16">
   <DarkCurrentValue CountOfArray="16">
     <item>5842.297852</item>
     <item>5683.556641</item>
     …
     <item>5113.907715</item>
   </DarkCurrentValue>
  </DarkCurrentData>
  <NoiseData VectorDim="16">
   <NoiseValue CountOfArray="16">
```

```
      <item>0.699997</item>
      <item>0.738245</item>
      …
      <item>0.367391</item>
    </NoiseValue>
  </NoiseData>
  <ToneCurvesData Row="33" Column="17">
    <CurveValue CountOfArray="561">
      <item>0.0</item>
      …
      <item>65535 </item>
      <item>0.0 </item>
      …
      <item>1.0</item>
    </CurveValue>
  </ToneCurvesData>
</InputDevData>

<InputImageInfo>
  <BandName VectorDim="16">
    <BandNameData>400nm 420nm … 700nm</BandNameData>
  </BandName>
  <IrisSetting VectorDim="16">
    <IrisSettingData>F2 F2.8 …F2</IrisSettingData>
  </IrisSetting>
  <ExposreTimeSetting VectorDim="16">
    <ExposreTimeSettingData>1/64 1/64 … 1/64</ExposreTimeSettingData>
  </ ExposreTimeSetting>
</InputImageInfo>

<InputIllu VectorDim="401" ShortWaveLength="380.0" DataNumber="401" WaveInterval="1.0">
  <InputSpecData CountOfArray="401">
      <item>0.007034</item>
      <item>0.007721 </item>
      …
      <item>0.167847</item>
  </InputSpecData>
</InputIllu>

< SubjectSpecMatrix>
  <EigenRefData Row="402" Column="10" ShortWaveLength="380.0" DataNumber="401"
                WaveInterval="1.0">
    <EigenRefValue CountOfArray="4020">
      <item>0.901000</item>
      <item>0.002343</item>
      …
      <item>0.823472</item>
    </EigenRefValue>
  </EigenRefData>
</SubjectSpecMatrix >

</NvisionInput>
```

## 5. "NvisionConversion" object

"NvisionConversion" object may contain three elements, "ColorConvData", "RenderingIllu" and "CMFData" as described in subsection 5.1 to 5.3.

Table 10 "NvisionConversion" object

| Element name | Data type | Description | Requirement level |
|---|---|---|---|
| ColorConvData | ColorConvDataType | Color conversion data ( for converting from one color space to another one ) [See Subsection 5.1] | OPTIONAL |
| RenderingIllu | RenderingIlluType | Rendering illuminant spectrum data [See Subsection 5.2] | OPTIONAL |
| CMFData | CMFDataType | A set of color matching functions data [See Subsection 5.3] | OPTIONAL |

### 5.1 "ColorConvData" element

"ColorConvData" element may contain the information related to the color conversion.

Table 11 "ColorConvData" element

| Element name | Data type | Description | Requirement level |
|---|---|---|---|
| SpecReflectData | SpecReflectDataType | Matrix data which is used to estimate normalized relative spectral reflectance values (0-1) from pixel values | OPTIONAL |
| SpecStimuliData | SpecStimuliDataType | Matrix data which is used to estimate absolute spectral radiance values [W/m$^2$/sr] from pixel values | OPTIONAL |
| XYZConvData | XYZConvDataType | Matrix data which is used to estimate the tristimulus values defined by an arbitrary set of color matching functions from pixel values | OPTIONAL |

The CIE1931XYZ tristimulus values X,Y and Z of the spectral radiance $s(\lambda)$ [W/m$^2$/sr] are calculated by the equation (3) using the CIE1931XYZ color matching functions $x(\lambda)$, $y(\lambda)$, and $z(\lambda)$, where the Y value is given in [cd/m$^2$] unit.

$$X = 683 \int_{\lambda=380}^{780} x(\lambda)s(\lambda)d\lambda$$
$$Y = 683 \int_{\lambda=380}^{780} y(\lambda)s(\lambda)d\lambda \qquad (3)$$
$$Z = 683 \int_{\lambda=380}^{780} z(\lambda)s(\lambda)d\lambda$$

The estimated spectral reflectance $\tilde{r}(\lambda_i)$, the estimated spectral radiance $\tilde{s}(\lambda_i)$ and the estimated tristimulus values $\tilde{X}\tilde{Y}\tilde{Z}$ are calculated by multiplying the M-band pixel values by the corresponding estimation matrix, the elements of which are described in "SpecReflectData", "SpecStimuliData" and "XYZConvData" respectively.
The "SpecReflectData", the "SpecStimuliData" and the "XYZConvData" consist of matrix elements shown in equation (4), (5) and (6) respectively.

$$\begin{bmatrix} \tilde{r}(\lambda_1) \\ \tilde{r}(\lambda_2) \\ \vdots \\ \tilde{r}(\lambda_R) \end{bmatrix} = \begin{bmatrix} m_{11}m_{12}\ldots m_{1M} \\ m_{21}m_{22}\ldots m_{2M} \\ \vdots \\ m_{R1}m_{R2}\ldots m_{RM} \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_M \end{bmatrix} \qquad (4)$$

$$\begin{bmatrix} \tilde{s}(\lambda_1) \\ s(\lambda_2) \\ \vdots \\ \tilde{s}(\lambda_R) \end{bmatrix} = \begin{bmatrix} m_{11}m_{12}\ldots m_{1M} \\ m_{21}m_{22}\ldots m_{2M} \\ \vdots \\ m_{R1}m_{R2}\ldots m_{RM} \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_M \end{bmatrix} \qquad (5)$$

$$\begin{bmatrix} \tilde{X} \\ \tilde{Y} \\ \tilde{Z} \end{bmatrix} = \begin{bmatrix} m_{11}m_{12}\ldots m_{1M} \\ m_{21}m_{22}\ldots m_{2M} \\ m_{31}m_{32}\ldots m_{3M} \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_M \end{bmatrix} \qquad (6)$$

### 5.1.1 SpecReflectData

The matrix elements $m_{ij}$ (i=1,…,R; j=1,…,M) which are used in the equation (4) to estimate the spectral reflectances $\tilde{r}(\lambda_i)$ at sampling wavelengths $\lambda_i=\lambda_0+(i-1)\Delta$ (i=1,…,R) from the pixel values $d_j$ (j=1,…,M) are described in the "SpecReflectValue"(ArrayOfFloatType) element,
where

$\Delta$ : the wavelength interval in nm,
$\lambda_0$ : the shortest wavelength in nm,
R : the number of sampling wavelengths.

The R and the M are described in the "Row" and the "Column" element of the "MatrixAttrGroup" attribute respectively.

The $\lambda_0$, the R (same R value described in the "Row" element of the "MatrixAttrGroup" attribute as mentioned above) and the $\Delta$ are described in the "ShortWaveLength", the "DataNumber" and the "WaveInterval" element of the "SpecDataAttrGroup" attribute respectively.

If the spectral reflectances are given as absolute values, "Absolute" is described in the "DEF" element of the "DataDefinitionAttrGroup" attribute. If the spectral reflectances are given as relative values, "Relative" is described in the "DEF" element of the "DataDefinitionAttrGroup" attribute. The spectral reflectance given as an absolute value is normalized so that the spectral reflectance of the perfect diffusing reflector is 1.0. If the illuminance is not regarded to be uniform on the object, "Relative" is described in the "DEF" element of the "DataDefinitionAttrGroup" attribute.

### 5.1.2 SpecStimuliData

The matrix elements $m_{ij}$ (i=1,…,R; j=1,…,M) which are used in the equation (5) to estimate the spectral radiances $\tilde{s}(\lambda_i)$ at sampling wavelengths $\lambda_i=\lambda_0+(i-1)\Delta$ (i=1,…,R) from the pixel values $d_j$ (j=1,…,M) are described in the "SpecStimuliValue"(ArrayOfFloatType) element,
where

$\Delta$ : the wavelength interval in nm,
$\lambda_0$ : the shortest wavelength in nm,
R : the number of sampling wavelengths.

The R and the M are described in the "Row" and the "Column" element of the "MatrixAttrGroup" attribute respectively.

The $\lambda_0$, the R (same R value described in the "Row" element of the "MatrixAttrGroup" attribute as mentioned above) and the $\Delta$ are described in the "ShortWaveLength", the "DataNumber" and the "WaveInterval" element of the "SpecDataAttrGroup" attribute respectively.

If the spectral radiances are given absolute values, "Absolute" is described in the "DEF" element of the "DataDefinitionAttrGroup" attribute. If the spectral radiances are given as relative values, "Relative" is described in the "DEF" element of the "DataDefinitionAttrGroup" attribute.

### 5.1.3 XYZConvData

The matrix elements $m_{ij}$ (i=1,2,3; j=1,…,M) which are used in the equation (6) to estimate the tristimulus values X, Y and Z from the pixel values $d_j$ (j=1,…,M) are described in the "XYZConvValue"(ArrayOfFloatType) element.

The number 3 and the M are described in the "Row" and the "Column" element of the "MatrixAttrGroup" attribute.

The identification information of the tristimulus values X, Y and Z are described in the "DATAID1", the "DATAID2" and the "DATAID3" element of the "DataAttrGroup" attribute respectively. When the tristimulus values X, Y and Z are the CIE1931XYZ tristimulus values, "CIE1931X", "CIE1931Y" and "CIE1931Z" are described respectively. When the tristimulus values X, Y and Z are the CIE1964XYZ tristimulus values, "CIE1964X", "CIE1964Y" and "CIE1964Z" are described respectively. When the tristimulus values are defined by the other method, the definition is described arbitrary way in the "DATAID1", the "DATAID2" and the "DATAID3" element of the "DataAttrGroup" attribute respectively.

If the trisitimulus values are given as absolute values satisfying the equation (3), "Absolute" is described in the "DEF" element of the "DataDefinitionAttrGroup" attribute. If the trisitimulus values satisfying the equation (3) are valid only relatively, "Relative" is described in the "DEF" element of the "DataDefinitionAttrGroup" attribute. If the trisitimulus values are neither the CIE1931XYZ tristimulus values nor the CIE1964XYZ tristimulus values, the "DEF" element should be "Relative".

## 5.2 “RenderingIllu” element

"RenderingIllu" element contains the information related to the rendering illuminant spectrum. The rendering illuminant is utilized to simulate the spectral radiance or the tristimulus values of the object from the estimated spectral reflectance or transmittance of the object. The rendering illuminant may or may not be identical to the input illuminant.

The spectral radiance of the object $\hat{s}(\lambda)$ is assumed to satisfy the equation (7) to be related to the spectral reflectance or transmittance of the object $r(\lambda)$ and the rendering illuminant spectrum $\hat{l}(\lambda)$ which is intended to illuminate the object.

$$\hat{s}(\lambda) = \hat{l}(\lambda) r(\lambda) \quad (7)$$

The sampled discrete data $\hat{l}(\lambda_i)$ (i=1,…,R) of the rendering illuminant spectrum $\hat{l}(\lambda)$ at sampling wavelengths $\lambda_i=\lambda_0+(i-1)\Delta$ (i=1,…,R) are described in the "RenderingSpceData"(ArrayOfFloatType) element, where

$\Delta$ : the wavelength interval in nm,
$\lambda_0$ : the shortest wavelength in nm,

R : the number of sampling wavelengths.

The number of sampling wavelengths R is described in the "VectorDim" element of the "VectorAttrGroup" attribute.

The $\lambda_0$ , the R (same R value described in the "VectorDim" element of the "VectorAttrGroup" attribute as mentioned above) and the $\Delta$ are described in the "ShortWaveLength", the "DataNumber" and the "WaveInterval" element of the "SpecDataAttrGroup" attribute respectively.

### 5.3 “CMFData” element

"CMFData" element contains the information related to a set of color matching functions.

The discrete data of the color matching functions $x(\lambda_i)$ , $y(\lambda_i)$ and $z(\lambda_i)$ at sampling wavelengths $\lambda_i=\lambda_0+(i-1)\Delta$ (i=1,…,R) are described in the "CMFValue"(ArrayOfFloatType) element,
where

$\Delta$ : the wavelength interval in nm,
$\lambda_0$: the shortest wavelength in nm,
R: the number of values.

The R and the number 3 are described in the "Row" and the "Column" element of the "MatrixAttrGroup" attribute respectively.

The $\lambda_0$ , the R (same R value described in the "Row" element of the "MatrixAttrGroup" attribute as mentioned above) and the $\Delta$ are described in the "ShortWaveLength", the "DataNumber" and the "WaveInterval" element of the "SpecDataAttrGroup" attribute respectively.

The identification information of the color matching functions $x(\lambda_i)$ , $y(\lambda_i)$ and $z(\lambda_i)$ are described in the "DATAID1", the "DATAID2" and the "DATAID3" element of the "DataAttrGroup" attribute respectively. When the color matching functions $x(\lambda_i)$ , $y(\lambda_i)$ and $z(\lambda_i)$ are the CIE1931XYZ color matching functions, "CIE1931X", "CIE1931Y" and "CIE1931Z" are described respectively. When the color matching functions $x(\lambda_i)$ , $y(\lambda_i)$ and $z(\lambda_i)$ are the CIE1964XYZ color matching functions, "CIE1964X", "CIE1964Y" and "CIE1964Z" are described respectively. When the color matching functions are defined by the other method, the definition is described arbitrary way in the "DATAID1", the "DATAID2" and the "DATAID3" element of the "DataAttrGroup" attribute respectively.

## 5.4 The XML schema definition of "NvisionConversion" object

The "NvisionConversion" object is defined by the XML schema as follows;

```
<xs:complexType name="NvisionConvType">
  <xs:sequence>
    <xs:element name="ColorConvData" type="d2p1:ColorConvDataType" minOccurs="0"/>
    <xs:element name="RenderingIllu" type="d2p1:RenderingIlluType" minOccurs="0"/>
    <xs:element name="CMFData" type="d2p1:CMFDataType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

```
<xs:complexType name="ColorConvDataType">
  <xs:sequence>
    <xs:element name="SpecReflectData" type="d2p1:SpecReflectDataType" minOccurs="0"/>
    <xs:element name="SpecStimuliData" type="d2p1:SpecStimuliDataType" minOccurs="0"/>
    <xs:element name="XYZConvData" type="d2p1:XYZConvDataType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

```
<xs:complexType name="SpecReflectDataType">
  <xs:sequence>
    <xs:element name="SpecReflectValue" type="d2p1:ArrayOfFloat"/>
  </xs:sequence>
  <xs:attributeGroup ref="d2p1:MatrixAttrGroup"/>
  <xs:attributeGroup ref="d2p1:SpecDataAttrGroup"/>
  <xs:attributeGroup ref="d2p1:DataDefinitionAttrGroup"/>
</xs:complexType>
```

```
<xs:complexType name="SpecStimuliDataType">
  <xs:sequence>
    <xs:element name="SpecStimuliValue" type="d2p1:ArrayOfFloat"/>
  </xs:sequence>
  <xs:attributeGroup ref="d2p1:MatrixAttrGroup"/>
  <xs:attributeGroup ref="d2p1:SpecDataAttrGroup"/>
  <xs:attributeGroup ref="d2p1:DataDefinitionAttrGroup"/>
</xs:complexType>
```

```
<xs:complexType name="XYZConvDataType">
  <xs:sequence>
    <xs:element name="XYZConvValue" type="d2p1:ArrayOfFloat"/>
  </xs:sequence>
  <xs:attributeGroup ref="d2p1:MatrixAttrGroup"/>
  <xs:attributeGroup ref="d2p1:DataAttrGroup"/>
  <xs:attributeGroup ref="d2p1:DataDefinitionAttrGroup"/>
</xs:complexType>
```

```
<xs:complexType name="RenderingIlluType">
  <xs:sequence>
    <xs:element name="RenderingSpecData" type="d2p1:ArrayOfFloat"/>
  </xs:sequence>
  <xs:attributeGroup ref="d2p1:VectorAttrGroup"/>
  <xs:attributeGroup ref="d2p1:SpecDataAttrGroup"/>
</xs:complexType>
```

```
<xs:complexType name="CMFDataType">
  <xs:sequence>
    <xs:element name="CMFValue" type="d2p1:ArrayOfFloat"/>
  </xs:sequence>
  <xs:attributeGroup ref="d2p1:MatrixAttrGroup"/>
  <xs:attributeGroup ref="d2p1:SpecDataAttrGroup"/>
  <xs:attributeGroup ref="d2p1:DataAttrGroup"/>
</xs:complexType>
```

## 5.5 Example of “NvisionConversion”

```
<NvisionConversion>

 <ColorConvData>
  <SpecReflectData Row="401" Column="16" ShortWaveLength="380.0" DataNumber="401"
                   WaveInterval="1.0" DEF="Relative">
   <SpecReflectValue CountOfArray="6416">
    <item>1.211</item>
    <item>0.923</item>
    …
    <item>0.002</item>
   </SpecReflectValue>
  </SpecReflectData>
  <SpecStimuliData Row="401" Column="16" ShortWaveLength="380.0" DataNumber="401"
                   WaveInterval="1.0" DEF="Relative">
   <SpecStimuliValue CountOfArray="6416">
    <item>0.99</item>
    <item>0.38</item>
    …
    <item>0.44</item>
   </SpecStimuliValue>
  </SpecStimuliData>
  <XYZConvData Row="3" Column="16" DATAID1="CIE1931X" DATAID2="CIE1931Y"
               DATAID3="CIE1931Z" DEF="Relative">
   <XYZConvValue CountOfArray="48">
    <item>0.12</item>
    <item>0.45</item>
    …
    <item>0.44</item>
   </XYZConvValue>
  </XYZConvData>
 </ColorConvData>

 <RenderingIllu d2p1:VectorDim="401" d2p1:ShortWaveLength="380.000000" d2p1:DataNumber="401"
                d2p1:WaveInterval="1.000000">
  <RenderingSpecData>2.1 3.5…1.0</RenderingSpecData>
 </RenderingIllu>

 <CMFData Row="401" Column="3" ShortWaveLength="380.0" DataNumber="401" WaveInterval="1.0"
          DATAID1="CIE1931X" DATAID2="CIE1931Y" DATAID3="CIE1931Z">
  <CMFValue CountOfArray="1203">
   <item>0.01</item>
   <item>0.02</item>
   …
   <item>0.00</item>
  </CMFValue>
 </CMFData>

</NvisionConversion>
```

## 6. Definition of basic data types

This document specifies the following basic data type.

ArrayOfFloatType

Floating point numbers $x_{11}$, $x_{12}$, …, $x_{1n}$, $x_{21}$, $x_{22}$,…, $x_{mn}$ are described. These numbers are elements of a matrix which has m rows and n columns. The data number which should be identical to m multiplied by n is described in the "CountOfArray" attribute. The number m and n are described in the "Row" attribute and the "Column" attribute of "MatrixAttrGroup" attribute group, respectively. When m=1, the number n may be described in the "VectorDim" attribute of "VectorAttrGroup" attribute group.

$$\begin{matrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \cdots & & & \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{matrix}$$

## 6.1 The XML schema definition of basic data types

A basic data type and attribute groups are defined by the XML schema as follows;

```
  <xs:complexType name="ArrayOfFloat">
    <xs:sequence>
      <xs:element name="item" type="xs:float" minOccurs="0" maxOccurs="unbounded" />
    </xs:sequence>
    <xs:attribute name="CountOfArray" type="xs:unsignedInt" use="required"/>
  </xs:complexType>
```

```
  <xs:attributeGroup name="VectorAttrGroup">
    <xs:attribute name="VectorDim" type="xs:unsignedInt" use="required"/>
  </xs:attributeGroup>
```

```
  <xs:attributeGroup name="MatrixAttrGroup">
    <xs:attribute name="Row" type="xs:unsignedInt" use="required"/>
    <xs:attribute name="Column" type="xs:unsignedInt" use="required"/>
  </xs:attributeGroup>
```

```
<xs:attributeGroup name="SpecDataAttrGroup">
    <xs:attribute name="ShortWaveLength" type="xs:float" use="required"/>
    <xs:attribute name="DataNumber" type="xs:unsignedInt" use="required"/>
    <xs:attribute name="WaveInterval" type="xs:float" use="required"/>
  </xs:attributeGroup>
```

```
  <xs:attributeGroup name="DataAttrGroup">
    <xs:attribute name="DATAID1" type="xs:string" use="required"/>
    <xs:attribute name="DATAID2" type="xs:string" use="required"/>
    <xs:attribute name="DATAID3" type="xs:string" use="required"/>
  </xs:attributeGroup>
```

```
  <xs:attributeGroup name="DataDefinitionAttrGroup">
    <xs:attribute name="DEF" type="xs:string" use="required"/>
  </xs:attributeGroup>
```

## 7. Implementation to JPEG2000

JPEG2000 is a set of lossless and lossy compression methods for digital image and the file format of the image using the method. Its basic part (JPEG 2000 Part1), which has a function to embed XML data, was recommended as an international standard in 2000[10]. The extended part (JPEG2000 Part2), which supports multi-channel image, was also recommended [11].
In this section the method to implement the NVXML data to the JPEG2000 file specified in the JPEG2000 Part2 (JPX file) is described.

A JPX file is a contiguous sequence of boxes as shown in figure 3. Each box is defined by a length and unique box type and contains the data defined for the box type. XML box that shall contain a well-formed XML document as data is optionally defined for the use of vender-specific information. A JPX file may not contain a XML box and can contain multiple XML boxes.

| | |
|---|---|
| JPEG 2000 Signature box | |
| File Type box | |
| Reader Requirements box | |
| JP2 Header box(superbox) | |
| | Label box |
| | Image Header box |
| | Bits Per Component box |
| | Colour specification box |
| | Palette box |
| | Component Mapping box |
| | Channel Definition box |
| | Resolution box |
| Codestream Header box(superbox) | |
| . . . | |
| **XML box** | |
| . . . | |

Figure 3. Conceptual structure of a JPX file

For enclosing the NVXML data in a JPX file, a XML Box that contains the NVXML data is added in the file. When using the NVXML data enclosed in a JPX file, identify XML boxes by the box type in the file and also identify the NVXML data by the signature "NVXML" described in the "NvisionImage" object in the XML boxes.

## 8. References

[1] W3C. Extensible Markup Language(XML) 1.0 (Second Edition) Rec-xml-20001006. October 2000
[2] W3C. Namespace in XML. Rec-xml-names-19990114. January 1999
[3] W3C. XML Schema Part0: Primer. PR-xmlschema-0-20010316. March 2001
[4] W3C XML Schema Part1: Structures. PR-xmlschema-1-20010316. March 2001
[5] W3C. XML Schema Part2: DataTypes. PR-xmlschema-2-20010316. March 2001
[6] W3C. Note, Date and Time Formats. August 1998
[7] CIE 15:2004 Colorimetry 3rd ed.
[8] IEC 61966-2-1:1999
[9] IEC 61966-2-4
[10] ISO/IEC 15444-1:2004
[11] ISO/IEC 15444-2:2004