ONVIF™
Imaging Service Specification

Version 18.12
December, 2018

CONTENTS

1 cope

2 Normative references

ONVIF Core Specification
<http://www.onvif.org/onvif/specs/core/ONVIF-Core-Specification.pdf>

3 Terms and Definitions

3.1 Definitions

Digital PTZ	Function that diminishes or crops an image to adjust the image position and ratio.
Imaging Service	Services for exposure time, gain and white balance parameters among others.
Input/Output (I/O)	Currently relay ports and video/audio inputs/outputs are handled.
Optical Zoom	Changes the focal length (angle of view) for the device by moving the zoom lens in the camera's optics.
Image Stabilization	Functionality used to avoid blurring of images due to movement of the device or its objects.
Tone Compensation	Functionality used to make the image with dark or bright areas to be more visible.
Defogging	Functionality used to make the image more detailed in presence of fog.
Imaging Preset	Functionality offered by Imaging device manufacturers as a tool to ease Image Setting adjustment for a given set of standard scene types.

4 Overview

The imaging service provides configuration and control data for imaging specific properties. WSDL is part of the framework and provided in the Imaging WSDL file.

The service includes the following operations:

- Get and set imaging configurations (exposure time, gain and white balance, for example).

- Get imaging configuration options (valid ranges for imaging parameters).

- Move focus lens including position and move status retrieval.

- Imaging Presets

WSDL for this service is specified in http://www.onvif.org/ver20/imaging/wsdl/imaging.wsdl.

Table 1: Referenced namespaces (with prefix)

Prefix	Namespace URI
env	http://www.w3.org/2003/05/soap-envelope
ter	http://www.onvif.org/ver10/error
xs	http://www.w3.org/2001/XMLSchema
tt	http://www.onvif.org/ver10/schema
timg	http://www.onvif.org/ver20/imaging/wsdl
tns1	http://www.onvif.org/ver10/topics

5 Service

The imaging service provides operations used to control and configure imaging properties on a device. A device that has one or more video sources should support the imaging service as defined in [ONVIF Imaging WSDL]. The imaging settings are part of the VideoSource entity. This means that imaging parameters directly affect a specific video source.

5.1 Imaging settings

5.1.1 Parameters

The imaging service provides operations to get or set imaging parameters and the valid ranges for those parameters. Some parameters have no effect if a specific mode is not set. Some of the parameters included in the settings require a specific imaging capability that can be requested through the GetOptions command. The following settings are available through the imaging service operations:

BacklightCompensation: Enables/disables BLC mode (on/off)

- On

    - Optional level parameter (unspecified unit).

- Off

Brightness: Adjusts the image brightness (unspecified unit).

ColorSaturation: Adjusts the color saturation in the image (unspecified unit).

Sharpness: Adjusts the sharpness in the image (unspecified unit).

Contrast: Adjusts the image contrast (unspecified unit).

Exposure:

- Auto – Enables the exposure algorithm on the device:

    - Priority – Sets the exposure priority mode (low noise/framerate).

    - Window – Rectangular exposure mask.

    - Min/MaxExposureTime – Exposure time range allowed to be used by the algorithm.

    - Min/MaxGain – The sensor gain range that is allowed to be used by the algorithm.

- o Min/MaxIris – The iris range allowed to be used by the algorithm.

- Manual – Disables the exposure algorithm on the device:

  - o ExposureTime – The fixed exposure time used by the image sensor (μs).

  - o Gain – The fixed gain used by the image sensor (dB).

  - o Iris – The fixed attenuation of input light affected by the iris (dB). 0dB maps to a fully opened iris.

Focus:

- Auto (parameters that apply to automatic mode only):

  - o Near/FarLimit – Limits for focus lens (m).

- Manual (parameters that apply to manual mode only):

  - o Default speed – The default speed for focus move operation (when the speed parameter not is present). Manual control is done through the move command, see Section 5.3.1.

IrCutFilter: Toggles the IR cut filter state between on, off and auto. The auto state lets the exposure algorithm handle when the Ir cut filter should be turned on or off.

- On: Enable IR cut fiter. Typically day mode.

- Off: Disable IR cut filter. Typically night mode.

- Auto: Ir cut filter is automatically activated by the device.

IrCutFilterAutoAdjustment: Optional parameters applied to only auto mode to adjust timing of toggling Ir cut filter

- BoundaryType – Specifies which boundaries to automatically toggle IR cut filter following parameters are applied to:

  - o Common: Applied to both boundaries automatically toggling IR cut filter on and off

  - o ToOn/ToOff: Applied individually to one of the boundaries automatically toggling IR cut filter to on/off

- BoundaryOffset – Adjusts boundary exposure level for toggling Ir cut filter to on/off specified with unitless normalized value from +1.0 to -1.0. Zero is default and -1.0 is the darkest adjustment (Unitless).

- ResponseTime – Delay time of toggling Ir cut filter to on/off after crossing of the boundary exposure levels

Whitebalance:

- Auto whitebalancing mode (auto/manual).

- Manual (parameters that apply to manual mode only):

  - o Rgain (unitless).

  - o Bgain (unitless).

WideDynamicRange: Whide dynamic range (on/off):

- On

- o Optional level parameter (unitless).

- Off

Image Stabilization: Enables/disables Image Stabilization feature (on/off/auto):

- On: Enable Image Stabilization feature.

  - o Optional level parameter (unspecified unit).

- Off: Disable Image Stabilization feature

- Auto: Image Stabilization feature is automatically activated by the device

ToneCompensation: Enables/disables Tone Compensation mode.

- On: Enable Tone Compensation feature.

  - o Optional level parameter – Optional level parameter to adjust level of compensation specified with normalized value from 0.0 to +1.0 (Unitless).

- Off: Disable Tone Compensation feature.

- Auto: Enable Tone Compansation feature with adjusting level automatically.

Defogging: Enables/disables Defogging function.

- On: Enable Defogging feature.

  - o Optional level parameter – Optional level parameter to adjust level of compensation specified with normalized value from 0.0 to +1.0 (Unitless).

- Off: Disable Defogging feature.

- Auto: Enable Defogging feature with adjusting level automatically.

Noise Reduction: Adjusts the noise reduction level in the image with normalized value from 0.0 to +1.0 (unitless). Level=0 means no noise reduction or minimal noise reduction.

5.1.2 GetImagingSettings

This operation requests the imaging setting for a video source on the device. A device implementing the imaging service shall support this command.

REQUEST:

- **VideoSourceToken [tt:ReferenceToken]**
  Reference to the video source for which the imaging settings are requested

RESPONSE:

- **ImagingSettings [tt:ImagingSettings20]**
  Imaging settings for the video source as defined in section 5.1.

FAULTS:

- **env:Sender - ter:InvalidArgVal - terNoSource**
  The requested VideoSource does not exist.

- **env:Receiver - ter:ActionNotSupported - ter:NoImagingForSource**
  The requested video source does not support imaging settings.

ACCESS CLASS:

> **READ_MEDIA**

If the video source supports any of the imaging settings as defined by the ImagingSettings type in the [ONVIF Schema], then it should be possible to retrieve the imaging settings from the device through the GetImagingSettings command.

5.1.3 SetImagingSettings

This operation sets the imaging settings for a video source on a device. A device implementing the imaging service shall support this command.

The possible configurable imaging settings parameters are described in section 5.1.1. Settings options are obtained through the command defined in section 5.1.4

REQUEST:

- **VideoSourceToken [tt:ReferenceToken]**
  Reference to the VideoSource

- **ImagingSettings [tt:ImagingSettings20]**
  The imaging settings to be applied

- **ForcePersistence - optional [xs:boolean]**
  Deprecated indicationwhether the configuration persists after reboot.

RESPONSE:

> This message is empty

FAULTS:

- **env:Sender - ter:InvalidArgVal - terNoSource**
  The requested VideoSource does not exist.

- **env:Receiver - ter:ActionNotSupported- terNoImagingForSource**
  The requested VideoSource does not support imaging settings.

- **env:Sender - ter:InvalidArgVal - terSettingsInvalid**
  The requested settings are incorrect.

ACCESS CLASS:

> **ACTUATE**

5.1.4 GetOptions

This operation gets the valid ranges for the imaging parameters that have device specific ranges. A device implementing the imaging service shall support this command. The command shall return all supported parameters and their ranges such that these can be applied to the SetImagingSettings command.

For read-only parameters which cannot be modified via the SetImagingSettings command only a single option or identical Min and Max values shall be provided.

REQUEST:

- **VideoSourceToken [tt:ReferenceToken]**
  Reference to the VideoSource

RESPONSE:

- **ImagingOptions [tt:ImagingOptions20]**
  Valid ranges for the imaging parameters that are categorized as device specific.

FAULTS:

- **env:Sender - ter:InvalidArgVal - terNoSource**
  The requested VideoSource does not exist.

- **env:Receiver - ter:ActionNotSupported- terNoImagingForSource**
  The requested VideoSource does not support imaging settings.

ACCESS CLASS:

READ_MEDIA

5.2 Imaging Presets

video source manufacturers can offer pre-set imaging configurations, which contain different parameter settings, tuned for optimum imaging performance under given scene circumstances. The Imaging Presets functionality is similar in function to what compact photography cameras offer today to the non-professional user, in order to easily be able to obtain optimum pictures under given scene conditions (e.g. Portrait, Landscape, Sports, Macro, Silouette).

Three commands enable the use of Imaging Presets.

5.2.1 GetPresets

The GetPresets command requests the current predefined list of Imaging Settings (Presets) offered by the manufacturer for a given video source.

The output is a list containing the available Imaging Presets. If the device offers no support for Imaging Presets, GetPresetsResponse shall return an empty list.

REQUEST:

- **VideoSourceToken**
  Reference token to VideoSource for which the available Imaging Presets should be requested.

RESPONSE:

- **Preset - unbounded [timg:ImagingPreset]**
  The available list of imaging presets supported by the device. Any device with Imaging Presets capability shall offer at least one Imaging Preset. In case the device offers no support for Imaging Presets, GetPresetsResponse shall return an empty list.

FAULTS:

- **env:Sender - ter:InvalidArgVal - terNoSource**
  The requested VideoSource does not exist.

ACCESS CLASS:

READ_MEDIA

5.2.2 GetCurrentPreset

The GetCurrentPreset command shall request the Imaging Preset which is currently applied to the specified video source, i.e. it shall request which of the predefined set of Imaging Settings (and/or Thermal Settings) was last applied to the video source. The output is the current Imaging Preset.

If Imaging Presets are not supported, or if the video source configuration does not match any of the existing Imaging Presets, the output of GetCurrentPreset shall be Empty.

REQUEST:

- **VideoSourceToken [tt:ReferenceToken]**
  Reference token to VideoSource for which the available Imaging Presets should be requested.

RESPONSE:

- **CurrentPreset - unbounded [timg:ImagingPreset]**
  The current (i.e. last applied) Imaging Preset applied to the specified video source.
  Return is empty if no capability or if current configuration does not match any existing
  Imaging Preset.

FAULTS:

- **env:Sender - ter:InvalidArgVal - terNoSource**
  The requested VideoSource does not exist.

ACCESS CLASS:

READ_MEDIA

5.2.3 SetCurrentPreset

The SetCurrentPreset command shall request a given Imaging Preset to be applied to the
specified video source. SetCurrentPreset shall only be available for video sources with
Imaging Presets capability.

Imaging Presets are defined by the Manufacturer, and offered as a tool to simplify Imaging
Settings adjustments for specific scene patterns. When the new Imaging Preset is applied by
SetCurrentPreset, as a response, the device shall adjust the video source settings to match
those values defined by the specified Imaging Preset.

REQUEST:

- **VideoSourceToken [tt:ReferenceToken]**
  Reference token to VideoSource for which the Imaging Presets should be set

- **PresetToken [tt:ReferenceToken]**
  Preset to be applied.

RESPONSE:

This message is empty

FAULTS:

- **env:Sender - ter:InvalidArgVal - terNoSource**
  The requested VideoSource does not exist.

ACCESS CLASS:

ACTUATE

5.3 Focus operations

5.3.1 Move

The move command moves the focus lens in an absolute, a relative or in a continuous manner
from its current position. The speed argument is optional for absolute and relative control, but
required for continuous. If no speed argument is used, the default speed is used. Focus
adjustments through this operation will turn off the autofocus. A device with support for
remote focus control should support absolute, relative or continuous control through the Move
operation. The supported MoveOpions are signalled via the GetMoveOptions command.

At least one focus control capability is required for this operation to be functional.

The move operation contains the following commands:

Absolute – Requires position parameter and optionally takes a speed argument. A unitless
type is used by default for focus positioning and speed. Optionally, if supported, the position
may be requested in m^{-1} units.

Relative – Requires distance parameter and optionally takes a speed argument. Negative distance means negative direction.

Continuous – Requires a speed argument. Negative speed argument means negative direction.

REQUEST:

- **VideoSourceToken [tt:ReferenceToken]**
  Reference token to the VideoSource.

- **Focus [tt:FocusMove]**
  Focal operation.

RESPONSE:

  This message is empty

FAULTS:

- **env:Sender - ter:InvalidArgVal - terNoSource**
  The requested VideoSource does not exist.

- **env:Receiver - ter:ActionNotSupported- terNoImagingForSource**
  The requested VideoSource does not support imaging settings.

ACCESS CLASS:

  ACTUATE

5.3.2 GetMoveOptions

The GetMoveOptions command retrieves the focus lens move options to be used in the move command as defined in Section 5.3.1. A device that supports the imaging service shall support the GetMoveOptions command. The response to the command shall include all supported Move Operations. If focus move is not supported at all, the reponse shall be empty.

REQUEST:

- **VideoSourceToken [tt:ReferenceToken]**
  Reference token to VideoSource.

RESPONSE:

- **MoveOptions [tt:MoveOptions20]**

  The valid ranges for the focus lens move command.

FAULTS:

- **env:Sender - ter:InvalidArgVal - terNoSource**
  The requested VideoSource does not exist.

- **env:Receiver - ter:ActionNotSupported- terNoImagingForSource**
  The requested VideoSource does not support imaging settings.

ACCESS CLASS:

  READ_MEDIA

5.3.3 Stop

The Stop command stops all ongoing focus movements of the lense. A device with support for remote focus control as signalled via the GetMoveOptions should support this command. The operation will not affect ongoing autofocus operation.

REQUEST:

- **VideoSourceToken [tt:ReferenceToken]**
  Reference token to the VideoSource.

RESPONSE:

> This message is empty

FAULTS:

- **env:Sender - ter:InvalidArgVal - terNoSource**
  The requested VideoSource does not exist.

- **env:Receiver - ter:ActionNotSupported- terNoImagingForSource**
  The requested VideoSource does not support imaging settings.

ACCESS CLASS:

> **ACTUATE**

5.3.4 GetImagingStatus

The GetStatus command requested the current imaging status from the device. A device with support for remote focus control as signalled via the GetMoveOptions shall support this command.

The imaging status contains:

- Focus position, move status and error information.

  - o The focus position is represented in a unitless type.

  - o Move status may be in a MOVING, IDLE or UNKNOWN state.

  - o Error information provided as a string, for example a positioning error indicated by the hardware.

REQUEST:

- **VideoSourceToken [tt:ReferenceToken]**
  Reference token to the VideoSource.

RESPONSE:

- **ImagingStatus [tt:ImagingStatus20]**

  The imaging status information.

FAULTS:

- **env:Sender - ter:InvalidArgVal - terNoSource**
  The requested VideoSource does not exist.

- **env:Receiver - ter:ActionNotSupported- terNoImagingForSource**
  The requested VideoSource does not support imaging settings.

ACCESS CLASS:

> **READ_MEDIA**

5.4 GetCapabilities

The capabilities reflect optional functions and functionality of a service. The information is static and does not change during device operation. The following capabilites are available:

ImageStabilization: Indicates whether or not Image Stabilization feature is supported.

ImagingPresets: Indicates whether or not Imaging Presets feature is supported.

REQUEST:

This message is empty

RESPONSE:

- **Capabilities [timg:Capabilities]**
  Set of indicators for function groups as described above.

FAULTS:

None

ACCESS CLASS:

PRE_AUTH

5.5 Events

The Message structure of these events is given by the following Message Description:

```
<tt:MessageDescription IsProperty="true">
  <tt:Source>
    <tt:SimpleItemDescription Name="Source" Type="tt:ReferenceToken"/>
  </tt:Source>
  <tt:Data>
    <tt:SimpleItemDescription Name="State" Type="xs:boolean"/>
  </tt:Data>
</tt:MessageDescription>
```

The SourceToken points to the source the image is coming from. This is in case of the Analytics or Image Service a VideoSource token and in case of the Recording Service the Recording job token.

5.5.1 Tampering

For video sources, Tamper Alarm situations are defined which indicate that the video camera is blended, covered, moved, or unplugged.

The tampering could be detected be different services (e.g by the Imaging Service or the Analytics Service). Some clients may want to get the information which service detected the event. Therefore the topic is used to give this additional information. The following three services are defined as source, a device MAY add additional ones if necessary.

5.5.1.1 ImageTooBlurry

When a camera is out-of-focus, such that important details are lost, the device should use the ImageTooBlurry notification in order to notify a client about the possible tampering if the device supports this feature.

```
tns1:VideoSource/ImageTooBlurry/AnalyticsService
```

```
tns1:VideoSource/ImageTooBlurry/ImagingService
```

```
tns1:VideoSource/ImageTooBlurry/RecordingService
```

5.5.1.2 ImageTooDark

When a camera is covered, such that the corresponding video signal becomes mostly black, the device should use the ImageTooDark notification in order to notify a client about the possible tampering if the device supports this feature.

```
tns1:VideoSource/ImageTooDark/AnalyticsService
```

```
tns1:VideoSource/ImageTooDark/ImagingService
```

```
tns1:VideoSource/ImageTooDark/RecordingService
```

5.5.1.3 ImageTooBright

When a camera is blended, such that the corresponding video signal becomes mostly white, the device should use the ImageTooBright notification in order to notify a client about the possible tampering if the device supports this feature.

```
tns1:VideoSource/ImageTooBright/AnalyticsService
```

```
tns1:VideoSource/ImageTooBright/ImagingService
```

```
tns1:VideoSource/ImageTooBright/RecordingService
```

5.5.1.4 GlobalSceneChange

When a large portion of the video content changes, the cause can be tamper actions like camera movement or coverage. If this feature is supported by the device it should notify the client using the following event:

```
tns1:VideoSource/GlobalSceneChange/AnalyticsService
```

```
tns1:VideoSource/GlobalSceneChange/ImagingService
```

```
tns1:VideoSource/GlobalSceneChange/RecordingService
```

5.5.1.5 SignalLoss

For devices having an analog input source, cutting the analog line between device and camera is a possible tampering which results in video Loss. A device that supports this feature shouldprovide the following event:

```
tns1:VideoSource/SignalLoss
```

5.5.2 MotionAlarm

When a device detects motion (e.g by an Analytics Service) it can inform a client using this event. This event is a basic motion alarm event that should be supported by all devices that support motion detection. If a device has a more complex algorithm running it is free to provide a vendor specific motion alarm event. If the device supports motion detection it should provide the following event.

```
tns1:VideoSource/MotionAlarm
```

Annex A. Revision History

Rev.	Date	Editor	Changes
2.1	Jul-2011	Hans Busch	Split from Core 2.0 without change of content.
2.1.1	Jan-2012	Hans Busch	Change Requests 535
2.2	Feb-2012	Takahiro Iwasaki	Addition of Image Stabilization and events Change Requests 654, 662, 664, 665
2.2.1	Dec-2012	Hans Busch	Change Request 708, 709
2.4	Mar-2013	Takahiro Iwasaki	Addition of IRCF Auto Adjustment parameters
2.5	Dec-2014	Takahiro Iwasaki Norio Ishibashi Hiroyuki Sano	Addition of ToneCompensation, Defogging and NoiseReduction parameters.
2.6.1	Dec-2015	Hans Busch	Change Request 1730
16.06	Jun-2016	Arsenio Vilallonga Hiroyuki Sano	Added Imaging Presets Change Request 1851
17.06	Jan-2017	Hans Busch	Updated method layouts. Change Request 1843
18.12	Dec-2018	Hiroyuki Sano	Change Request 2361