Archived:Symbian Onboard Camera Advanced Settings API

Archived: This article is archived because it is not considered relevant for third-party developers creating commercial solutions today. If you think this article is still relevant, let us know by adding the template {{ReviewForRemovalFromArchive|user=~~~~|*write your reason here*}}.

Note: This API is not part of the public SDK. It can be found in the SDK API Plug-in.

Purpose

`CCameraAdvancedSettings` class, part of the Symbian Onboard Camera (`CCamera`) API, is used for controlling advanced settings of the camera hardware. On S60 platform, `CCameraAdvancedSettings` is supported from S60 3rd Edition, Feature Pack 2 onwards.

Use cases

`CCameraAdvancedSettings` provides methods for controlling aperture size, autofocus and focus range, bracketing, drive mode (single shot, burst, continuous) and various other settings. Detailed CCameraAdvancedSettings API reference and guide can be found in the Library.

Note that camera hardware used in S60 devices does not support all the functionality defined in `CCameraAdvancedSettings`. The API provides functions for querying supported settings for each advanced camera feature.

One important use case is controlling the autofocus in devices with AF-lens camera module. Prior to S60 3rd Edition, Feature Pack 2, autofocus support for camera applications was enabled with an add-on autofocus plug-in library, part of S60 Platform: Camera Example with AutoFocus Support. This autofocus plug-in is not supported anymore on S60 3rd Ed, FP2 devices. The example application below demonstrates how to control autofocus on S60 3rd Ed, FP2 devices using `CCameraAdvancedSettings`.

Example code

Headers:

```
#include <ecam.h>
#include <ecamadvancedsettings.h>
```

Link against:

```
LIBRARY ecam.lib ecamadvsettings.lib
```

Constructing CCamera and CCameraAdvancedSettings:

`CCamera` instance is constructed with `New2L` to enable all events from the camera hardware. The observer for the camera (here, *this* class) must implement `MCameraObserver2` interface.

```
// Use camera index 0 (the main camera) with priority 0
iCamera = CCamera::New2L( *this, 0, 0 );
iAdvancedSettings = CCamera::CCameraAdvancedSettings::NewL( *iCamera );

// Proceed to reserve and power on the camera
// ...
```

Controlling advanced camera settings:

After the camera has been successfully reserved and powered on, advanced settings can be accessed through `iAdvancedSettings`. For example, focus range can be set as follows:

```
// Set focus range to macro
iAdvancedSettings->SetFocusRange(CCamera::CCameraAdvancedSettings::EFocusRangeMacro);
```

Changing any advanced camera settings will always result in a call to `MCameraObserver2::HandleEvent()`:

```
void CMyCameraEngine::HandleEvent( const TECAMEvent &aEvent )
  {
  if( aEvent.iEventType == KUidECamEventCameraSettingFocusRange2 )
    {
    if( aEvent.iErrorCode == KErrNone )
      {
```


(C) Copyright Nokia 2013. All rights reserved.

```
        // Autofocus range successfully set
        }
      else
        {
        // Error occurred (request focus range not supported, camera not ready, etc...)
        }
      }
    ...
    }
```

See other available functions, UID values and enumerations in *ecamadvancedsettings.h*.

Note: Some of the event types are deprecated and should be ignored (`KUidECamEventCameraSettingFocusRange`, `KUidECamEventCameraSettingAutoFocusType`) - the correct event type to handle are `KUidECamEventCameraSettingFocusRange2` and `KUidECamEventCameraSettingAutoFocusType2`).

Example Application

Below is a version of S60 Platform: Camera Example with AutoFocus Support, modified to use `CCameraAdvancedSettings`. Event handling is done by implementing `MCameraObserver2` interface. Due to these changes, the example can only be run on S60 3rd Edition, Feature Pack 2 devices.

Camera Example with AutoFocus Support for S60 3rd Edition, Feature Pack 2

(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.