Image Filter

Programming Guide

Version 1.0

Table of Contents

1. Overview

Image Filter offers you easy image processing and provides you with various types of image filters. It allows you to apply various effects and filter presets such as Sepia, Pastel, and Cartoonization to an image.

You can use the Image Filter package to:

- apply a range of filter effects
- adjust filter levels
- adjust image transparency

The following figure shows the results when a filter and filter level are applied to an image.

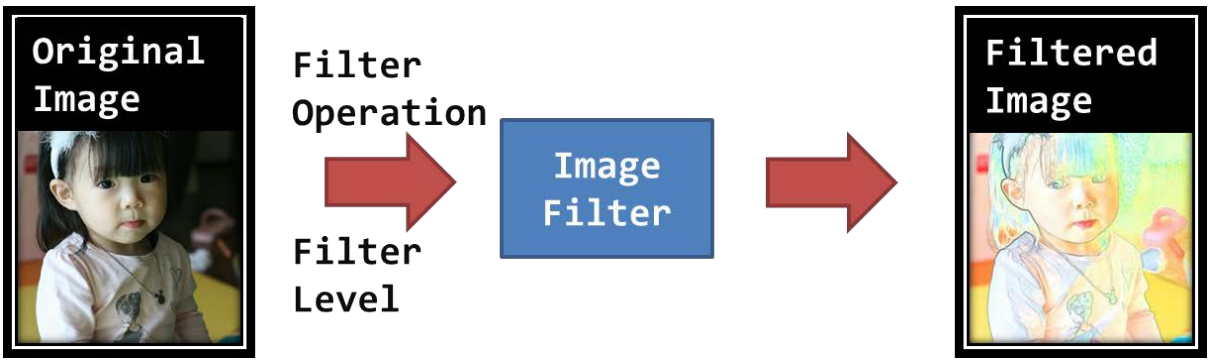

Figure 1: Filter and filter level applied to an image

1.1. Architecture

The following figure shows the Image Filter architecture.

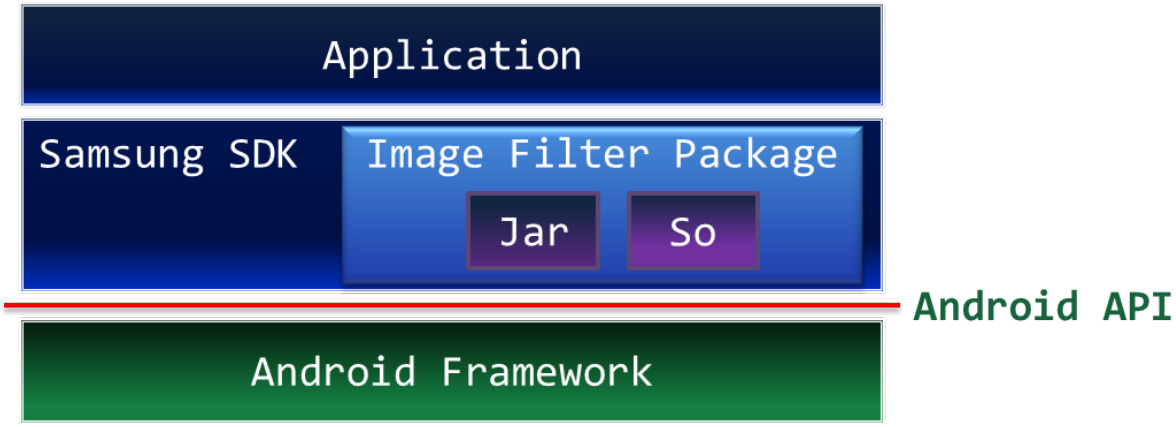

Figure 2: Image Filter architecture

The architecture consists of:

- **Applications:** One or more applications that use the Image Filter package.
- **Image Filter Package:** Image Filter components built on top of the Android Framework. The Image Filter methods are static and you can call them without instantiating the Image Filter classes.

1.2. Class Diagram

The following figure shows the Image Filter classes that you can use in your application.

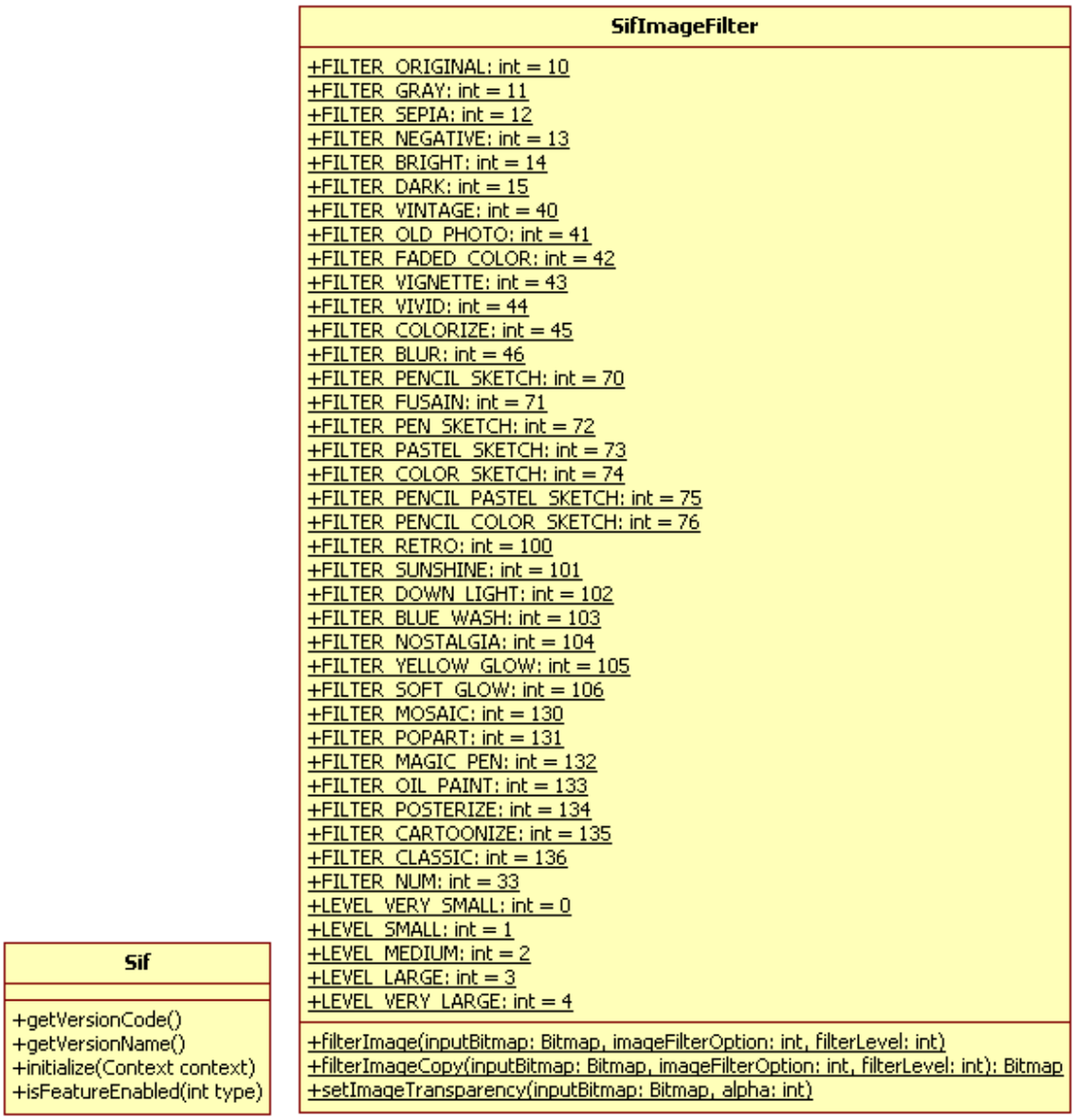

Figure 3: Image Filter classes

The Image Filter classes include:

- **Sif:** Initializes the Image Filter package.
- **SifImageFilter:** The main class of the Image Filter package, which provides static functions for managing image filters and image transparency.

For more information on the Image Filter classes, see the Image Filter API Reference.

1.3. Supported Platforms

Android 4.0 (Ice Cream Sandwich API level 14) or higher support Image Filter.

1.4. Supported Features

The Image Filter package supports the following features:

- 33 image filters
- filter level adjustment
- transparency adjustment

1.5. Components

- Components:
  - imagefilter-v1.0.0.jar
  - libImageFilterLibs.so
- Imported packages:
  - com.samsung.android.sdk.imagefilter

1.6. Installing the Package for Eclipse

To install Image Filter for Eclipse:

1. Add the imagefilter-v1.0.0.jar, libImageFilterLibs.so, and sdk-v1.0.0.jar files to the libs folder in Eclipse.

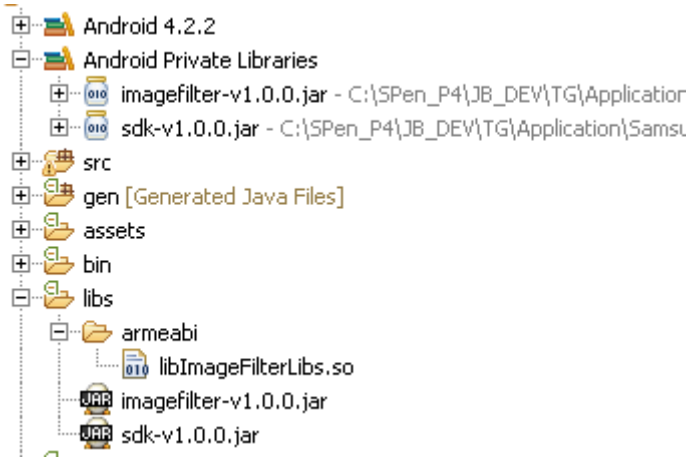

Figure 4: libs folder in Eclipse

2. Hello Image Filter

Hello Image Filter is a sample program that:

- initializes the Image Filter package
- gets an image
- applies a filter

```
// HelloImageFilter.java
public class Hello extends Activity
{
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

       Sif imageFilter = new Sif();
       try {
            // Initialize the Sif instance.
            imageFilter.initialize(this);
      } catch (SsdkUnsupportedException e) {
          // Error Handling
      }

      mVersionCode = imageFilter.getVersionCode();
      mVersionName = imageFilter.getVersionName();

      // Get the original image
      Bitmap backgroundBitmap = BitmapFactory.decodeResource(getResources(),
               R.drawable.baby);

       // Apply image filter
       Bitmap filteredBitmap = SifImageFilter.filterImageCopy(
           backgroundBitmap, SifImageFilter.FILTER_SEPIA,
           SifImageFilter.LEVEL_MEDIUM);

}
```

3. Sif Class

You need to initialize a Sif before you can use it. Samsung Mobile SDK provides a base class with an initialize() method for each package.

The Sif can run only on Samsung Smart Devices. Some Samsung Smart Device models do not support some *of the* packages.

You can use a `initialize()` method to initialize it and also to check if the device supports the Sif. If the device does not support the Sif, the method throws an SsdkUnsupportedException exception. You should handle this exception. If an SsdkUnsupportedException exception is thrown, you can check the exception type with **`SsdkUnsupportedException.getType()`**. If the device is not a Samsung device, the exception type is **SsdkUnsupportedException.VENDOR_NOT_SUPPORTED**. If the device is a Samsung model that does not support the Sif, the exception type is **SsdkUnsupportedException.DEVICE_NOT_SUPPORTED**.

The Sif class provides the following methods:

- `initialize()` initializes Image Filter.
- `getVersionCode()` gets the Image Filter version number as an integer.
- `getVersionName()` gets the Image Filter version name as a string, which starts from “1.0”.

```
Sif imageFilter = new Sif();
try {
    // Initialize the Sif instance.
    imageFilter.initialize(this);
} catch (SsdkUnsupportedException e) {
    // Error Handling
}

mVersionCode = imageFilter.getVersionCode();
mVersionName = imageFilter.getVersionName();
```

3.1. Using the Initialization Method

The initialization method provides the following functionality:

- Initializes the package.
- Checks if the device is a Samsung Device.
- Checks if the device supports the package.
- Checks if the package libraries are installed on the device.

```
void initialize(Context context) throws SsdkUnsupportedException
```

If the method fails to initialize the Sif, it throws an **SsdkUnsupportedException** exception**.** If an exception is thrown, you should check the exception message type.

3.2. Ssdk UnsupportedException Messages

SsdkUnsupportedException exceptions are generated if the initialization method fails to initialize the Sif. The class defines the following exception types:

- **VENDOR_NOT_SUPPORTED:** The device is not a Samsung device.
- **DEVICE_NOT_SUPPORTED:** The device is a Samsung device that does not support the package.

4. Using the Image Filter Package

The Image Filter methods are static, which means you can use them without instantiating them.

4.1. Applying Image Filters

The Image Filter package provides two methods for applying filters to an image:

- `filterImage()`: Applies the filter to the original image.
- `filterImageCopy()`: Copies the image, applies the filter, and returns the filtered image.

The parameters for both these methods include the original image, the image filter value (defined in the class), and the value for the filter level (defined in the class).

```
// Apply image filter to original image
SifImageFilter.filterImage(backgroundBitmap, SifImageFilter.FILTER_SEPIA,
                SifImageFilter.LEVEL_MEDIUM);

// Apply image filter to a copy and return result
Bitmap resultBitmap = SifImageFilter.filterImageCopy(
                backgroundBitmap, SifImageFilter.FILTER_GRAY,
                    SifImageFilter.LEVEL_SMALL);
```

4.2. Adjusting Transparency

You can use the `setImageTransparency()` method for setting the image transparency. This method applies the alpha value to the original image. The alpha value ranges from 0 to 255. When the alpha value increases, the opacity also increases, which means the transparency decreases.

Figure 5: Adjusting image transparency

```
int alphaValue = 100;
// set alpha value
    SifImageFilter.setImageTransparency(backgroundBitmap, alphaValue);
```

Copyright

Though every care has been taken to ensure the accuracy of this document, Samsung Electronics Co., Ltd. cannot accept responsibility for any errors or omissions or for any loss occurred to any person, whether legal or natural, from acting, or refraining from action, as a result of the information contained herein. Information in this document is subject to change at any time without obligation to notify any person of such changes.

Samsung Electronics Co. Ltd. may have patents or patent pending applications, trademarks copyrights or other intellectual property rights covering subject matter in this document. The furnishing of this document does not give the recipient or reader any license to these patents, trademarks copyrights or other intellectual property rights.

No part of this document may be communicated, distributed, reproduced or transmitted in any form or by any means, electronic or mechanical or otherwise, for any purpose, without the prior written permission of Samsung Electronics Co. Ltd.

The document is subject to revision without further notice.

All brand names and product names mentioned in this document are trademarks or registered trademarks of their respective owners.

For more information, please visit http://developer.samsung.com/