Nokia Imaging SDK和Aviary SDK的简单介绍

大家都知道随着1020的发布，NOKIA特别推出了NOKIA IMAGE SDK，以方便开发者对照片和图片的处理，特别是在1020这样的怪兽级别的机器上；过往我们对图片的处理都是使用第三方的SDK，Aviary SDK。过往也是非常好用，就是里面会有其品牌信息，有一些不爽，现在NOKIA IMAGE SDK的推出应该会更加方便了，这次特别对这2个SDK做一下简单的使用WIKI，若各位喜欢可以继续跟帖，我们会继续更新更多的深入使用WIKI

Introduction

安装 Nokia Imaging SDK可以到这里填写相关信息后下载SDK进行安装，也可直接在Visual Studio中使用NuGet搜索Nokia Imaging SDK安装到项目即可。 Aviary SDK可以到这里下载到最新版本，安装后即可使用。

Nokia Imaging SDK的使用 首先我们需要引入以下命名空间：

```
using Nokia.Graphics.Imaging;
using System.Runtime.InteropServices.WindowsRuntime;
using Microsoft.Phone.Tasks;
using Windows.Storage.Streams;
using Microsoft.Xna.Framework.Media;
```

然后我们通过选择器选择图片，并对选择的图片流进行处理：

```
private EditingSession _session;
private void PickAnImageButton_Click(object sender, RoutedEventArgs e)
{
    PhotoChooserTask chooser = new PhotoChooserTask();
    chooser.Completed += PickImageCallback;
    chooser.Show();
}
private async void PickImageCallback(Object sender, PhotoResult e)
{
    if (e.TaskResult != TaskResult.OK)
    {
        return;
    }
    _session = await EditingSessionFactory.CreateEditingSessionAsync(e.ChosenPhoto);
    try
    {
        // Decode the jpeg for showing the original image
        await _session.RenderToImageAsync(OriginalImage,
OutputOption.PreserveAspectRatio);
        // Apply the filter and decode into the bitmap.
        _session.AddFilter(FilterFactory.CreateCartoonFilter(true));
        await _session.RenderToImageAsync(CartoonImage,
OutputOption.PreserveAspectRatio);
    }
    catch (Exception exception)
    {
        MessageBox.Show("Exception:" + exception.Message);
        return;
    }
    SaveButton.IsEnabled = true;
}
```

最后进行保存操作：

(c) Copyright Nokia 2014. All rights reserved.

```
private async void SaveImage_Click(object sender, RoutedEventArgs e)
{
    if (_session == null)
    {
        return;
    }
    // The image has been already filtered.
    // We only need to render the output as a jpeg
    IBuffer jpegOut = await _session.RenderToJpegAsync();
    // Save the image as a jpeg to the camera roll
    MediaLibrary library = new MediaLibrary();
    string filename = "SavedPicture_" + DateTime.Now.ToString("G");
    Picture pic = library.SavePicture(filename, jpegOut.AsStream());
    MessageBox.Show("Image saved");
}
```

Aviary SDK的使用 首先要将下面的代码复制到App.xaml中：

```
xmlns:resources="clr-namespace:AviarySDK.Resources;assembly=AviarySDK"

    <!--Application Resources-->
    <Application.Resources>
        <ResourceDictionary>
            <ResourceDictionary.MergedDictionaries>
                <ResourceDictionary Source="/AviarySDK;component/Themes/generic.xaml"/>
            </ResourceDictionary.MergedDictionaries>
            <resources:Images x:Key="ImageResources"/>
            <aviary:LocalizedStrings xmlns:aviary="clr-
namespace:AviarySDK;assembly=AviarySDK" x:Key="LocalizedStrings" />
        </ResourceDictionary>
</Application.Resources>

然后我们通过选择器选择图片，并对选择的图片流进行处理：
<code csharp>
private void PickAnImageButton_Click(object sender, RoutedEventArgs e)
{
    PhotoChooserTask chooser = new PhotoChooserTask();
    chooser.Completed += PickImageCallback;
    chooser.Show();
}
private async void PickImageCallback(Object sender, PhotoResult e)
{
    if (e.TaskResult != TaskResult.OK)
    {
        return;
    }
    AviaryTask aviaryTask = new AviaryTask(e.ChosenPhoto, AviaryFeatrue(), themeColor:
"32A9FF", originalFileName: "imgii", maxInputMegaPixels: 8, maxOutputResolution:
GetMaxSizeByQuality((Application.Current as App).PicQuality));
            aviaryTask.Completed += new
EventHandler<AviaryTaskResultArgs>(aviaryTask_Completed);
            aviaryTask.Show();
}

加载要应用的效果：
```


(c) Copyright Nokia 2014. All rights reserved.

```
<code csharp>
private List<AviaryFeature> AviaryFeatrue()
        {
            List<AviaryFeature> features = new List<AviaryFeature>();
               features.Add(AviaryFeature.Enhance);
features.Add(AviaryFeature.Effects);            features.Add(AviaryFeature.Orientation);
features.Add(AviaryFeature.Crop);
            features.Add(AviaryFeature.Text);
            features.Add(AviaryFeature.Blemish);
            features.Add(AviaryFeature.Brightness);
            features.Add(AviaryFeature.Contrast);
            features.Add(AviaryFeature.Draw);
            features.Add(AviaryFeature.Redeye);
            features.Add(AviaryFeature.Saturation);
            features.Add(AviaryFeature.Sharpness);
               features.Add(AviaryFeature.Stickers);
features.Add(AviaryFeature.Whiten);
            //features.Add(AviaryFeature.About);
            return features;
        }
```

保存Aviary SDK处理后得到的图片：

```
void aviaryTask_Completed(object sender, AviaryTaskResultArgs e)
        {
            if (e.AviaryResult == AviaryResult.OK)
            {
                //this.mainCanvas.Background = e.PhotoResult;
                WriteableBitmap we = e.PhotoResult;
                Stream stream = new MemoryStream();
                we.SaveJpeg(stream, we.PixelWidth, we.PixelHeight, 0, 100);
            }
            else
            {
                //aviaryTask_Error(e.Exception);
            }
        }
```


(c) Copyright Nokia 2014. All rights reserved.

(c) Copyright Nokia 2014. All rights reserved.