LUMIA APP LABS #14

WORKING WITH HIGH-RESOLUTION IMAGES

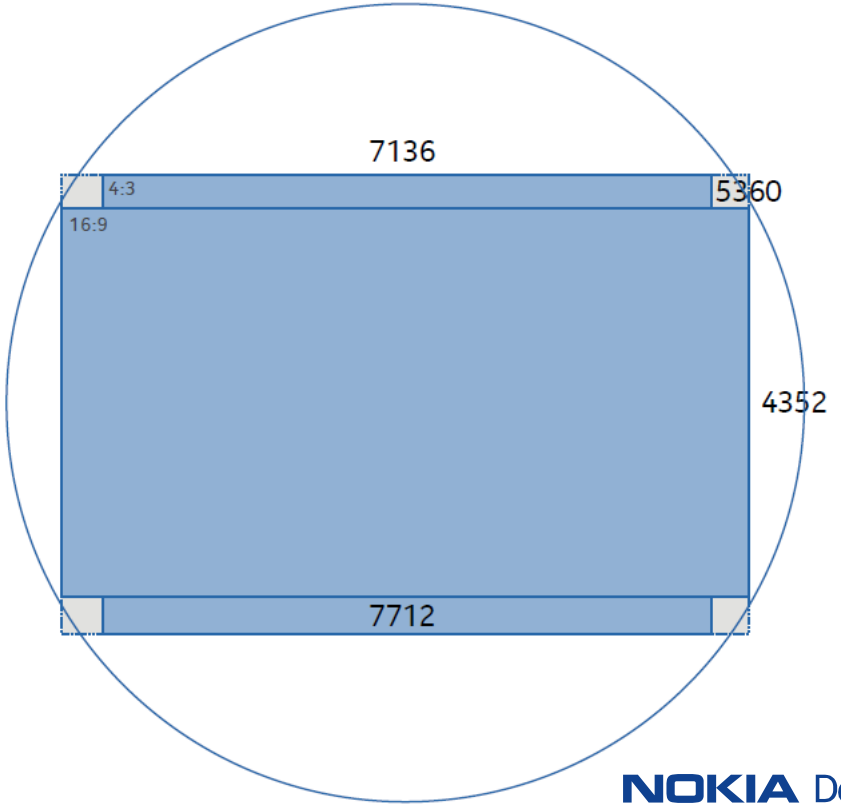
Berthier Lemieux
Technology Wizard

NOKIA Developer

NOKIA LUMIA 1020 CAMERA SENSOR

Sensor size 1/1.5”
Pixel size 1.1 micron
Optical Image Stabilization

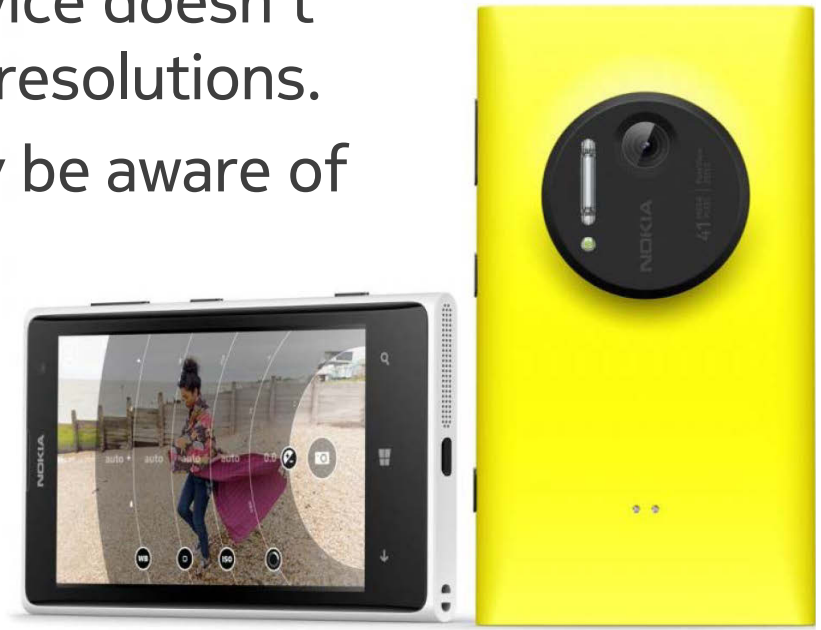
Total sensor : 7712x5360 = 41.3MP
16:9 mode : 7712x4352 = 33.6 MP
4:3 mode : 7136x5360 = 38.2 MP



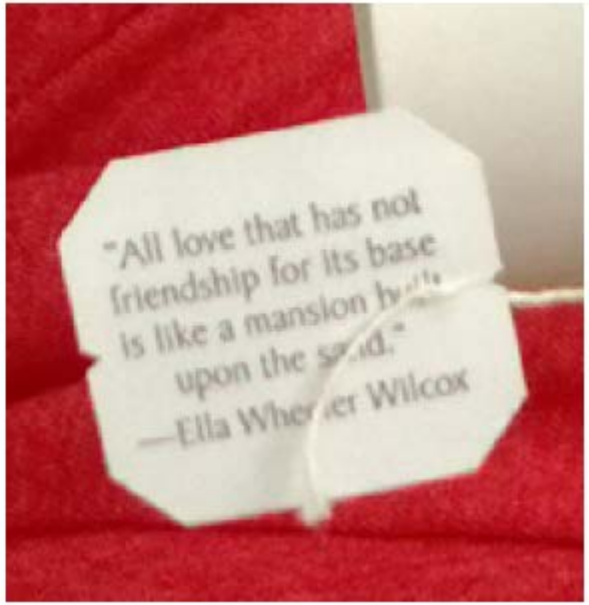
5 MP OR HIGH RES?

- By default, the PhotoCaptureDevice doesn't expose the 34 or 38 megapixels resolutions.
- Applications needs to specifically be aware of that resolution.

Without any changes, your app will capture 5 MP images and they will look awesome!



ZOOMING


"All love that has not
friendship for its base
is like a mansion
upon the
—Ella

BE NICE TO YOUR NEIGHBOURS!

High res images requires special attention and may break other applications.

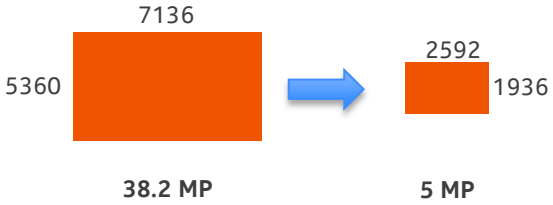
Don’t save full resolution images to the camera roll.

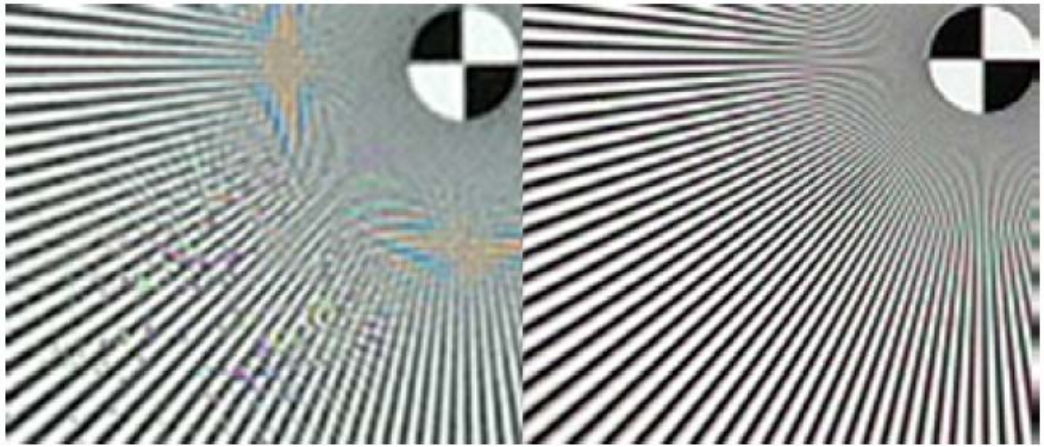
DUAL CAPTURE IN NOKIA CAMERA PRO

The camera application saves both the full resolution and an oversampled image simultaneously.

OVERSAMPLING

Use the data contained in the full resolution image to create the best 5MP image.





Standard 5 MP

Oversampled 5 MP

DUAL CAPTURE IN NOKIA CAMERA PRO

- The high resolution image is saved **locally** to the application.
- The high resolution image file, the ”original”, will never be modified. That file can’t be accessed through the PhotoPickerTask.
- An oversampled (5 MP) image is saved to the camera roll.
- Whenever they desire, the user can reframe the image; change where and how deep they want to zoom in on the image. A new oversampled (5 MP) image is saved to the camera roll, overwriting the existing one.
- It is highly recommended to follow a similar design pattern in your application.

SAMPLE APP: PHOTO INSPECTOR

[Download the code from developer.nokia.com]

STEPS

1. Capture a high resolution image.
2. Save it to your local storage.
3. Do your magic.
4. Scale down the resulting image to 5MP.
5. Save the 5MP image to the camera roll.
6. Implement the Rich Media extension.
7. Keep application local storage clean.

STEP 1 : CAPTURE A HIGH RESOLUTION IMAGE

```
captureResolution = PhotoCaptureDevice.GetAvailableCaptureResolutions(SENSOR_LOCATION).First();
var deviceName = Microsoft.Phone.Info.DeviceStatus.DeviceName;
 if (deviceName.Contains("RM-875") || deviceName.Contains("RM-876") || deviceName.Contains("RM-877"))
    {
     // We now knows that the application runs on a Lumia 1020
       captureResolution = new Windows.Foundation.Size(7712, 4352);    // 16:9 ratio
     //captureResolution = new Windows.Foundation.Size(7136, 5360);   // 4:3 ratio
     }
  _device = await PhotoCaptureDevice.OpenAsync(SENSOR_LOCATION, captureResolution);
```

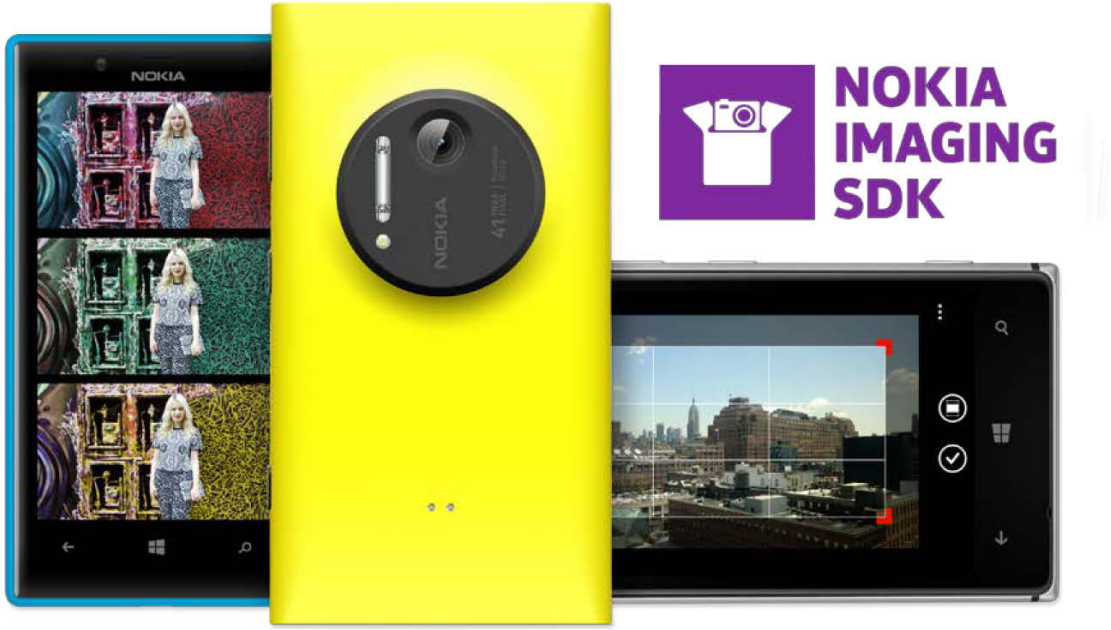
STEP 2 : SAVE HIGH RES TO LOCAL STORAGE

```
using (var store = IsolatedStorageFile.GetUserStoreForApplication())
{
   var localPath = @"\LocalImages";
   if (!store.DirectoryExists(localPath))
      {
         store.CreateDirectory(localPath);
      }
   using (var file = store.CreateFile(localPath + @"\" + filenameBase + @".jpg"))
   {
      using (var localImage = image.AsStream())
      {
         localImage.CopyTo(file);
         file.Flush();
      }
   }
}
```

STEP 3 : DO YOUR MAGIC

STEP 4: SCALE DOWN THE RESULTING IMAGE TO 5MP

This is easy to do, using the Nokia Imaging SDK.



STEP 5 : SAVE THE 5MP IMAGE TO CAMERA ROLL

```
var library = new Microsoft.Xna.Framework.Media.MediaLibrary())
  {
       using (var picture = library.SavePictureToCameraRoll(filenameBase, libraryImage))
        {
           savedPath = picture.GetPath();
        }
    }
  }
```

STEP 6: IMPLEMENT THE RICH MEDIA EXTENSION

Register your app as a rich media application (in WMAppManifest.xml)

When launched, the NavigationContext.QueryString of your application will contain the 2 keys:

{[Action, RichMediaEdit]}

{[token, {9b6bb83d-18db-45d3-9ab1-44b7a80ff2ab}]}]

In your OnNavigatedTo, match the photo coming from the media library files with the locally saved photo

MATCH LOCAL FILE AND MEDIA LIBRARY FILE

```
protected override void OnNavigatedTo(NavigationEventArgs e)
  {   …
      var queryString = this.NavigationContext.QueryString;
      if (queryString.ContainsKey("token"))
        {
            using (Microsoft.Xna.Framework.Media.Picture picture = library.GetPictureFromToken(token))
            {
                var localPath = picture.GetPath();
                /* … Some code removed here : Parse the path to retrieve the filename */
                using (var store = IsolatedStorageFile.GetUserStoreForApplication())
                {
                    if (store.FileExists(localPathCandidate))
                    {
                        return store.OpenFile(localPathCandidate, FileMode.Open);
                    }
            }
```

STEP 7 : KEEP APPLICATION LOCAL STORAGE CLEAN

- High resolution images are huge in size : Offer a way for users to delete the images you're keeping in the local storage.
- It is a good practice to automatically check if the image has been deleted from the media library.

SUMMARY

Your application will capture amazing images on the Nokia Lumia 1020!

If you decide to capture high res images:

- **Follow the code sample from the “Photo Inspector”.**
- **Be nice with your other third-party applications: don’t save high resolution images to camera roll.**
- **Use the Nokia Imaging SDK for scaling and cropping operations.**

RESOURCES

- Imaging in the Lumia Developer's Library: http://nokia.ly/WP_lib_img
- Photo Inspector example application: http://nokia.ly/PhotoInspector
- Nokia Imaging SDK: http://www.developer.nokia.com/imaging
- Nokia Imaging discussion board: http://nokia.ly/DiBoImg
- Lumia 1020 White paper: http://nokia.ly/N1020WhiteP
- Windows Phone: Building a Camera App: http://channel9.msdn.com/Events/Build/2013/2-210
- Rich media extensibility for Windows Phone 8: http://msdn.microsoft.com/en-us/library/windowsphone/develop/jj662942(v=vs.105).aspx

THANK YOU!

NOKIA IMAGING WIKI COMPETITION 2013Q3

Write a great tutorial, guide or an article with code which shows how to use the Nokia Imaging SDK or Camera and any other Windows Phone APIs related to imaging in useful, imaginative and innovative ways and/or provide the best feedback on the Nokia Imaging SDK.

developer.nokia.com/Community/Wiki/

NOKIA Developer