

Nokia Imaging and Big UI Wiki Competition 2013Q4



Competition closed (24 Dec 2013): Check out [the winners](#) and other great [entries](#) below.

The Winners

With 22 entries, most of them having a lot of useful, innovative and interesting technical information, picking our winners was difficult. In the end we awarded an additional [Lumia 1520](#). We also picked a small number of developers for honourable mention - they won't get a phone, but like the winners they will get 500 [DVLUP](#) XP points to reflect their technical achievement. While we've deliberately restricted mention to those articles that we rated very highest, many of our other [entries](#) had several nominations and are well worth learning from.

Without further ado, the winners are

Imaging article winners:

- SB Dev for [Creating and optimizing a custom effect for the Nokia Imaging SDK](#)
This article shows how to create and benchmark the same custom effect with the Nokia Imaging SDK using C#, C++, and ARM NEON SIMD instructions. The guidance means that developers can understand their options, and at what point optimising using C++ and ARM NEON might be worth further investigation. SB Dev has also been supportive of other developers on the wiki and in the discussion boards during the competition.
- pkovacevic for [Creating a simple and beautiful Mosaic effect for Nokia Imaging SDK 1.0](#)
This effect uses a clever approach to create a visually and computationally effective mosaic effect. This is also a great example of how to create a custom effect that takes multiple source images for processing.
- MaMi for [Tiltshift effect using Nokia Imaging SDK](#)
 - The Tiltshift or "Miniature Faking" custom filter to make images look like they are photographs of miniature scenes is a really stunning effect.
- igrali for [Augmented reality for fun using Nokia Imaging SDK filters](#)
 - A good explanation of GART and augmented reality, but what inspired many of the judges was the opportunities offered by being able to apply imaging SDK filters over the augmented reality camera feed.
- galazzo for [Image processing optimization techniques](#)
 - This article outlines a number of useful image processing techniques. The content is more suitable to advanced developers wishing to work with C++ and ARM NEON.

"Big UI" article winners:

- to_pe for [Advanced Techniques for Big UI](#)
This article provides a number of techniques for maintaining XAML, graphical and other resources to support all the Windows Phones screen sizes and resolutions. The judges particularly liked the final approach, which allows developers to specify XAML only for those screens that would really benefit, rather than the traditional approach where a screen needs to be created for every screen resolution.
- Tomas Slavicek for [UI Framework for XNA and MonoGame with high DPI support](#)
XNA/Monogame developers will like this straightforward approach for defining a UI that will scale properly for all supported WP screen resolutions.

The following articles have also received an honourable mention. While we don't have prizes, we recognise the technical effort and the authors will get 500XP points if registered with DVLUP:

- [All About Custom Filters with the Nokia Imaging SDK](#) (Rob.Kachmar) - We particularly liked the instructions in "Setup", which allow you to start coding a custom filter in no time without worrying about infrastructure.
- [Custom Effects in Nokia Imaging SDK: Pixelate and Color Quantizer](#) (RatishPhilip) - Very nice explanation of custom filters, along with some good examples.
- [Layout on Large Screens](#) (MMlejnek) - we loved the TileCanvas control - takes the effort out of using tiles on large devices.
- [Semantic Zoom in Windows Phone 8](#) (dtimotei) - very cool control, favoured by many judges

Feedback winner

The best feedback came from SB Dev , who will receive a Nokia Lumia 925. yan_ receives an "honourable" mention both as "runner up" in the feedback section, but also for his update to [Optimizing Imaging SDK use for rapidly changing filter parameters](#), and for his community spirit in providing excellent feedback to other authors in the competition

Thank you to everyone who competed. Please join me in congratulating the winners on the [discussion boards](#)

Background

We're holding a wiki competition to celebrate the release of [Nokia Imaging SDK 1.0](#) and the launch of the "Phabulous" new [Lumia 1520](#) and [Lumia 1320](#) devices.

The competition has two main focus areas. Firstly, we are looking for great original articles, tutorials, and guides that show how to use the features of the updated Nokia Imaging SDK with both C++ and C# APIs. Secondly, we're looking for articles which provide new insight and guidance on how to best use the [Lumia 1520](#)'s increased screen resolution and size (and of course its excellent imaging capabilities).

Prizes for winning wiki articles will be a [Lumia 1520](#) device - we'll be awarding up to 6 competitors. A 7th winner will receive a [Lumia 925](#) for the best [feedback](#) we get about the updated Nokia Imaging SDK.

 App or Wiki competition?: Win by contributing **original articles** in this wiki competition or by submitting innovative imaging **apps** to the [Nokia Create Image & Photo Master Mission](#). You can do even both - find out how [below](#)!



Getting Started

For imaging article entries, the best way to get to know the features of the SDK is to [download](#) it and start coding.

If you're familiar with the beta SDK, you should start by reading [How to upgrade](#).

If not, the [Lumia Developer's Library](#) comes with great documentation and

examples for using the Imaging SDK, camera, and working with high resolution images. Useful resources include

- [Nokia Imaging SDK documentation](#):
 - [Quick Start](#)
 - [Nokia Developer Example Projects](#)
 - [Filter Effects](#) - using different filters with camera photos and saving in the camera roll album.
 - [Filter Explorer](#) - explore image editing capabilities and performance by applying filter layers to images.
 - [Real-time Filter Demo](#) - Apply effects in real-time to the stream received from the camera and shown in the viewfinder.
 - [API reference](#)
 - [Using the library with C++](#)
- [Imaging](#) (Lumia Developer's Library)
 - [Advanced Photo Capturing](#)
 - [Camera Explorer](#) (Example)
 - [Working with high resolution photos](#)
 - [Photo Inspector](#) (Example)
- [Lumia App Labs](#)
 - [Episode 4: Creating stunning imaging apps](#)
 - [Episode 14: Working with high resolution images on Windows Phone 8](#)
 - [Episode 15: Using the Nokia Imaging SDK](#)

If you're aiming to provide an entry about working with the bigger screen sizes and resolution, the following are essential viewing:

- [Optimising for large screen phones](#) (Nokia Lumia Developer's Library)
- [Dynamic Layout](#) sample (Nokia Lumia Developer's Library)
- [Multi-resolution apps for Windows Phone 8](#) (Dev Center)

- [Nokia Music Explorer](#) sample (Nokia Lumia Developer's Library) was re-factored to scale for the new screen resolution.
- [Lumia App Labs: Episode 17: Optimising apps for large screens](#)
- [Taking Advantage of Large-Screen Windows Phones](#) (Windows Phone Developer Blog)

How to participate

Write a great tutorial, guide, or article with code that shows how to use the Nokia Imaging SDK 1.0 in useful, imaginative, and innovative ways, OR write an original and useful article about using the new screen size.

Before creating an article you should review the content in the wiki, libraries and SDK and consider whether your article would provide new and useful information. Articles to check include those from the [Q3 imaging competition](#), and in the wiki categories: [Camera on Windows Phone](#), [Imaging on Windows Phone](#) and [Video on Windows Phone](#).

The competition will run until the 15th December, giving you just over a month.

Here are the competition rules:

- To enter, add a link to your article as a comment to this article.
- The article and associated [Code Example](#) are submitted under the site [Terms & Conditions](#).
- Articles must be written in English. Translations may be provided in other languages, but only the English version will be judged.
- The article should include the text: **{{Note|This is an entry in the [[Nokia Imaging and Big UI Wiki Competition 2013Q4]].}}**
- You'll be able to win one prize only, regardless of the number of articles you write.
- Teamwork is allowed and even endorsed, but only one prize will be granted to a winning team (this will go to the person who *creates* the article initially). Teams may also receive additional complementary Nokia products.
- All articles must be submitted by 23:59 PST on 15 December 2013.

See the [full terms and conditions](#) for more information.

The prizes

The six "wiki article winners" will receive a [Lumia 1520](#) device. Broadly speaking we hope to award 4 prizes for articles about using the Imaging SDK C++ or C# API, and 2 prizes for articles that focus on any aspect of working with the larger screen size/resolution. We reserve the right to allocate prizes differently based on the overall evaluation of entries in each category.

A seventh winner will receive a [Lumia 925](#) for the best [feedback](#) we get about the updated Nokia Imaging SDK.

In addition, winning a Nokia Developer Competition gives you instant fame and kudos in the community and promotion through Nokia Developer channels. It also looks really great on your CV.

Evaluation of entries

The primary criteria entries will be judged on are the relevance, usefulness, quality, and presentation of their technical information. Multiple entries are permitted, but will be assessed on their individual merits — a single great entry will win over multiple articles that aren't as good.

The entries will be judged by the [Wiki Moderation team](#) and selected Nokia technical experts.

Feedback

Feedback for the Nokia Imaging SDK 1.0 should be posted in the [Windows Phone Imaging](#) discussion board, and can cover any aspect of the SDK (including the documentation and example code), and both problems and suggestions for improvement.

To make it easier for us to respond properly to feedback, contributors should post individual threads. So that we can identify competition entries, please use the following title syntax for your feedback post: "Imaging SDK 1.0 Feedback Competition: <userid>: <short description of feedback>" (for example "Imaging SDK 1.0 Feedback Competition: hamishwilliee : New API is a lot more complicated").

Generic questions or comments concerning this competition should be raised as comments to this wiki page.

Stuck for ideas?

Article entries should demonstrate innovative use or combination of features, address real world use cases or provide comprehensive tutorial instruction. They might also highlight tips, tricks, and best practices for working with features and APIs.

[Optimising for large screen phones](#) and [Dynamic Layout](#) sample provide a comprehensive overview for using larger screen resolutions/sizes. If you're competing in this area finding a unique idea will be challenging - consider complementary articles which provide concrete examples of where the advice in the above article and example work.

If you're planning to write about the possibilities provided by the [Nokia Imaging SDK 1.0](#) (and articles about imaging "in general"), we've put some ideas below:

- Articles about features which are better on high resolution Lumia devices - like the [Lumia 1020](#) and [Lumia 1520](#)
- Porting Nokia Developer Examples to use C++ APIs and do basic benchmarking
- Articles that show how to use the Imaging SDK and camera functions with C++
- Articles that address use cases which can be done only with C++ (e.g. using NEON, or porting custom effects/filters written in C++ from other platforms).
- Real time filtering on video feeds, using the C++ API
- Enhanced camera and video photography - enhancing existing photography through new features. For example, adding a "horizon" overlay, or arbitrarily setting exposure or focus points through the camera UI
- Auto correction of images, image filters and effects applied as the picture/video is taken or applied as a post process, addition of drawing or commenting.
- Image visualisation and presentation - approaches for arranging, presenting and merging groups of photos or videos that go beyond the native device gallery app.
- Show how to recreate "classic" filters using the SDK, combine them in innovative ways, and also implement new filters not covered by the SDK
- Imaging sharing and storing innovations - through social media, remote storage like DropBox or SkyDrive, NFC. Making it easy to "plug-in" to new services.
- Cost effective data transfer - including ideas like better compression, sharing only when using low cost carriers like WLAN
- Monetization aspects - making money through imaging apps, selling mobile photos, using advertising effectively for imaging apps. Embedding advertisements in images.
- Camera or image based games and augmented reality
- Printing from device
- How to use imaging in "real world contexts", including for medical purposes, warranty claims, construction.
- Locally relevant imaging solutions - for example using imaging solutions to identify and track local animal species or dangerous areas.

In addition, the [Ideas Project](#) has hundreds of app ideas related to imaging and photography - many of which might also inspire great wiki articles.

Competition entries

All entries will be listed below as they are received.

- [Green Screen Magic \(Chroma Key\) with Nokia Imaging SDK](#) (summeli)
- [Filter and Custom Filter Management Framework for the Nokia Imaging SDK](#) (Engin Kirmacı)
- [Edge detection filter in Windows Phone](#) (somnathbanik)
- [Creating and optimizing a custom effect for the Nokia Imaging SDK](#) (SB Dev)
- [UI Framework for XNA and MonoGame with high DPI support](#) (Tomas Slavicek)
- [All About Custom Filters with the Nokia Imaging SDK](#) (Rob.Kachmar)
- [Advanced Techniques for Big UI \(to_pe \)](#)
- [Tiltshift effect using Nokia Imaging SDK](#) (MaMi)
- [Porting Existing Apps to large Screen Device](#) (SANDEEP SHARMA)
- [Creating a simple and beautiful Mosaic effect for Nokia Imaging SDK 1.0](#) (pkovacevic)
- [Simulate 1080p windows phone emulator](#) (rudyhuyn)
- [Augmented reality for fun using Nokia Imaging SDK filters](#) (igrali)
- [All About Designing apps for Big Screen Windows Phone](#) (chintandave_er)
- [Layout on Large Screens](#) (MMlejnek)
- [9 Patch Image Alternative For Windows Phone](#) (vinayppatil)
- [Nokia Imaging SDK in native code](#) (galazzo)
- [Placement Of UI Elements For various screen types](#) (ApoorvUpadhyay)
- [Semantic Zoom in Windows Phone 8](#) (dtimotei)

- [Custom Effects in Nokia Imaging SDK: Pixelate and Color Quantizer](#) (RatishPhilip)
- [Creating an edition history with undo and redo actions optimized for Nokia imaging SDK](#) (juan_K)
- [Image processing optimization techniques](#) (galazzo)

Article updates - may be eligible for consideration.

- [Optimizing Imaging SDK use for rapidly changing filter parameters](#) (yan_)
- [Filter Parameter Harmonization: Dynamic UI for Nokia Imaging SDK 1.0 Filters](#) (RatishPhilip)

Nokia Create: Image and Photo Master Mission



[Nokia Create](#) is a global competition to develop apps for Nokia Lumia and Windows Phone 8. There are 15 Missions, built around 8 categories, and there will be up to 151 winners. The Image & Photo Master Mission invites developers to create a brand new imaging app, or port one from another platform to Windows Phone 8. Prizes for winners include trips to Mobile World Congress, devices, [DVLUP](#) experience points (XPs), promotions, credit for ad campaigns and much more.

You can compete in both competitions: enter the *Nokia Create Image & Photo Master Mission* by submitting new apps, and take part in the *Nokia Imaging and Big UI Wiki Competition 2013Q4* by contributing original wiki articles. As long as your wiki article code example hasn't been published on the store it can also be evolved into a proper app and submitted to Nokia Create.

Copyright ©2013 Nokia. All rights reserved. Nokia and Nokia Developer are registered trademarks of Nokia Corporation.