Case Study: 4tiitoo Constructs a Modern User Interface with Voice, Gesture, and Eye Tracking Input

By Karen Marcus

In 2012, Intel held the Europe, Middle East, and Africa-based Ultrabook™ Experience Software Challenge to encourage developer invention and imagination in enabling a more spontaneous user experience with Ultrabook devices. Thirty participants from 11 countries competed for 6 weeks to develop original applications that integrated touch, gesture, and voice functionality. Judging criteria were as follows:

- **Functionality.** Does the application work quickly and effectively, without any issues?
- **Creativity.** Does the application represent an innovative usage model?
- **Commercial potential.** How useful is the application for the mass market?
- **Design.** Is the application simple to understand and easy to use?
- **Fun factor.** How positive is the emotional response to the application?
- **Stability.** Is the application fast and simple, without glitches?

The software company 4tiitoo (pronounced “forty-two”), as a participant in the Ultrabook Experience Software Challenge, designed the winning app, NUIA* Imagine, a photo organizing and viewing application running on the Windows* 8 desktop.

With a focus on natural user experience, the development team sought to use a variety of input modalities that offer a more comfortable computing experience than the traditional keyboard and mouse. Although the functionality of the app is familiar, the way the user interacts with it is unusual, with touch, gesture, and voice input as well as an eye-tracking component. The result is a modern user interface (UI) that allows multiple types of input for the same commands. For example, a hand swipe to the right, pressing the right arrow key, or saying, “Next,” all result in the app showing the image to the right of the current image on the screen.

**Product**

The idea for NUIA Imagine came from the 4tiitoo team and the problems they faced in organizing their photos from vacations and other events. The task was neither pleasurable nor efficient for them, and they decided to address the dilemma with an application that provides more flexible, yet intuitive functionality. The team developed NUIA Imagine specifically for the Ultrabook Experience Software Challenge.

NUIA Imagine enables users to organize images into albums. The application reads all images within specified directories and displays them as thumbnails in the Miniature Preview in the lower-right area of the screen. The Workbench, in the center, displays thumbnails in a larger resolution, including a center image, which users can add to the active album. Any number of
albums can be created, and users can switch among them using the Album Overview in the upper-left area of the screen. The Album Preview, in the upper-right area of the screen, displays the selected album (see Figure 1).

![NUIA Imagine interface](image)

**Figure 1. NUIA Imagine interface**

NUIA Imagine is unique not because of what users can do with it but for how they can do it. Silke Eineder, marketing manager at 4tiitoo, explains: “Users can organize or enjoy looking at their photos in the most comfortable way. Instead of sitting in front of a computer for hours in an uncomfortable position, bound to a mouse and keyboard, NUIA Imagine makes use of Ultrabook sensors to allow users to organize photos from a relaxed position. This is possible because they can work hands-free, as NUIA Imagine supports hand gestures and voice commands.”

So, users can literally sit back and have a cup of coffee while using the application. They can simply go through their photos, delete the ones that did not turn out well, organize the others into separate folders, or view the photos just by speaking various commands such as, “add,” “delete,” “next,” “previous,” and “maximize” (see Figure 2). Keystrokes and touch gestures can alternately be used to perform the same functions.
Development Process for Windows* 8

The development team decided to create NUIA Imagine as a Windows 8 Desktop application so it could also be used in older Windows versions and be easily ported to other operating systems. To ensure operation on other operating systems, says Eineder, “Except for touch, we did not use any key features of Windows 8. We did this to support other versions of Windows and to make sure we didn’t have problems porting the application to non-Microsoft operating systems.”

Another challenge was that the speech recognition software included with Windows 8 was less accurate than the software the team had used in the past. So the solution they found was the Nuance VoCon speech engine, which demonstrated much better recognition performance. Eineder explains: “The software needs a grammar file with the commands. On any speech input, it delivers a result with the top three recognitions and the respective detection rates. The basic design of the application and the used parts of the NUIA software development kit (SDK) libraries define all actions (swipe, add, delete, etc.) completely agnostic from any trigger event. So, every modality, such as speech, only needs to send a trigger event (the top recognized command). Everything else is already implemented in the lower stack. The speech recognition engine, which is used in the Intel® Perceptual Computing SDK, is similar to this engine.”
Development Tools

The team used the multimodal NUIA SDK, the Qt Creator, and many sheets of paper to develop NUIA Imagine.

NUIA SDK

The NUIA Imagine application was developed based on the NUIA SDK and middleware, which Stephan Odörfer, co-founder and CTO, describes: “The NUIA SDK is a hardware-agnostic infrastructure for connecting all kinds of input modalities to standard operating system interactions. That means there’s an abstraction layer inside that sends out a command, like, ‘Next.’ This command can be triggered by any modality in use by the computer. For example, a ‘Next’ command could be input using the right cursor key, a swipe gesture, or voice input. The design of the software is completely agnostic with these modalities. So, a programmer could, without any problem, add another modality, which then already controls the actions of the software, without any further development effort.

“The NUIA SDK also connects to other SDKs, such as the Intel Perceptual Computing SDK. It allows the creation of multisensor-optimized applications and the enhancement of legacy applications without the need for in-depth sensor knowledge.”

Another modality used within NUIA Imagine is eye tracking. Odörfer notes, “Eye tracking is a very important modality in the NUIA SDK, and we worked very closely with several divisions within Intel and also Tobii Technologies from Sweden to implement it.”

The following list, adapted from the 4tiitoo website, outlines other functions of the NUIA SDK:

- The NUIA tools provide integrated development environment wizards, debug tools, and the Extension Creator, a graphical UI tool, to create multimodal extensions for legacy applications, without the need for any source code modification.
- The NUIA user experience provides a powerful set of libraries, application programming interfaces (APIs), and bindings to several programming languages and frameworks.
- The NUIA Core provides a message-passing infrastructure for its plug-ins and a control UI.
- The NUIA Core plug-ins contain the main functionalities and communicate over well-defined messages (with a maximum of abstraction in mind), connect to various SDKs and low-level APIs for retrieving input data, generate legacy events (e.g., keyboard shortcuts, mouse cursor control), and can also implement more complex algorithms and macros.
- The Interprocess Communication Framework assures communication between the NUIA components and NUIA-enhanced applications.
- The Context subsystem provides information about all states of the underlying operating system (e.g., currently focused application, user logged in, and screen resolution).
- The NUIA documentation provides a comprehensive set of tutorials, examples, and support tools.
**Qt Framework**

The team had worked previously with the Qt framework and therefore was familiar with its capabilities. Qt is an event-driven framework: touch events are embedded in the framework, and they can be recognized and handled just like mouse or keyboard events. This functionality provided the team with the ability to create the application such that it can respond to touch events just as it reacts to mouse events.

**Development Process for Ultrabook Platforms**

NUIA Imagine supports several input modalities, including keyboard, mouse, touch screen, speech, gestures, and eye tracking. These modalities offer users a faster and more immersive experience. The team determined which modalities to include based on case discussions of typical user situations.

**Touch and Gestures**

As part of the focus on natural user experience, the team incorporated touch and 3D hand gestures. Eineder explains, “Touch and gestures are more natural to humans because these actions are part of our daily interaction with other people and things. Human eye-hand coordination is optimized for these kinds of movements (such as swiping left or right), rather than pressing different keys.”

NUIA Imagine supports touch by using Qt-based touch events. To determine which touch optimizations to make, the team analyzed which touch gestures were most intuitive to use with the application and were already known to users based on their experience with smartphones and tablets. They tested the optimizations with people not involved in the development process.

One optimization was improving the recognition of swipe gestures. The challenge inherent with this modification, says Eineder, is, “Every user performs gestures a little differently; however, the application needs to recognize all of them.” She adds, “We did a lot of testing regarding the time span after which a gesture is recognized. After that, we fine-tuned the variables responsible for the detection process. This adjustment made the touch feature much more intuitive to use.” The variables are used to define the time between the “touch begin,” “touch update,” and “touch end.” The correlation among the three was fine-tuned and user tested for more accurate touch recognition.

As another input modality, 3D gestures are used to control the main functionalities. The 3D gestures supported within NUIA Imagine are swipe left for next image, swipe right for previous image, and swipe up to add the current image to the active album. These three gestures come from the OpenNI* software. Other gestures are possible, but given the time parameters of the challenge, the team decided to implement just those three.
Gestures are recognized in the application with the NUIA SDK. Bastian Wolfgruber, chief application developer, says, “We use OpenNI to track the gestures, and then wire the NUIA Core. The gesture commands are sent to the application, which reacts to those gestures.” He adds, “There’s no need to calibrate. The user just holds up an arm; when it is recognized, the user can do the gestures.”

**Voice Recognition**

NUIA Imagine uses the speech-recognition software from Nuance. All major interactions can be triggered by speech. The application recognizes seven voice commands: “next,” “previous,” “add,” “delete,” “minimize,” “maximize,” and “rotate.” The team wanted the voice modality to be simple and intuitive, so users could begin interacting with the application without reading application documentation.

To arrive at the decision to use voice recognition, the team discussed different possibilities for the main use case. Odörfer says, “Using voice recognition is an elegant way to command an application without actually sitting directly in front of the computer. Speech is a natural communication, like gestures or eye movements, in contrast to the standard existing technologies like the keyboard or the mouse.”

The voice-recognition modality currently operates only in English. However, the application is set up to be multilanguage. Odörfer observes, “To extend the languages, we would only need to implement new dictionary files because we use a speech-recognition engine. Using the Nuance Framework, you just add, for example, a German dictionary file, and then the application also reacts to German commands.”

**Eye Tracking**

In addition to keyboard and mouse, touch, gestures, and voice recognition, NUIA Imagine can be controlled via eye tracking. Odörfer comments, “Eye tracking is an important modality in the NUIA SDK.”

The application allows eye tracking to indicate which element the user wants to interact with. Eye tracking can also be combined with voice commands and other modalities. For example, a user could look at any picture in a gallery, say, “Add,” and the picture is added to the album.

As another example, if the user looks at the Workbench, three images are available: the main picture, the previous image, and the next image. If the user looks at the next image, it automatically moves to the main position. However, says Odörfer, “Auto-gaze actions are not always intended. For example, if you look at a Delete button, you might not want to trigger the action immediately. So, in most cases, the user performs the triggering action with a specific key on the keyboard, the middle mouse button, or other intentional triggering.”

Odörfer adds, “Eye tracking will not completely replace other modalities, but in combination with other modalities, it greatly enhances work on a next-generation computer.”
Eineder notes, “It makes it more comfortable because you do not have to use a keyboard or mouse. You can choose, and you can lay back and relax. On the business side, it’s more productive because you can look at a menu and open it while keeping your hands on the keyboard.”

Eye tracking is intuitive for users. Odörfer says, “It takes most users maybe half a minute or a minute, and then they completely adapt. At first, they think they would have to look differently from the way they normally do, but actually users just look as they always look at their screen, and the system performs an action without the user touching anything.” The team performed user testing to ensure that typical users would understand the eye tracking actions as an appropriate response from the application. Odörfer says, “The idea is to support the user, and there is an advantage to using eye tracking for this.”

An eye tracking peripheral and the NUIA Software Suite must be installed to use the eye-tracking function. When the application is launched for the first time, the user must do a 30-second calibration to enable eye tracking. Odörfer says, “The current generation of eye tracking has a level of accuracy of 0.5 degrees, which is, in a standard operating mode, something like 15 or 20 pixels on the monitor, similar to touch screen accuracy. So, users cannot control small buttons, which are used in some desktop applications, but they can easily control applications that are optimized for touch screens or Windows Store applications because the buttons are large enough.

In the NUIA SDK, we have components that understand which element is below or maybe close by, and then click this element, even if the exact gaze position is not on the element. This is similar to using an Android* or iPad* tablet and touching a browser link but not hitting it exactly. The browser checks to see if there is a link close by. If there is a link, it activates the closest one to the touch point.”

As a demonstration of this technology, 4tiitoo partnered with Intel and Tobii to enable the game Minecraft* from Mojang to be controlled with eye tracking using the native NUIA SDK parts. This version of the game was presented in the Intel booth at MINECON 2012.

Eineder comments, “In general, with eye tracking, there are a whole lot of possibilities that will come up. For example, you can easily control the Windows 8 Start screen with it. As soon as this technology is made available in Ultrabook [devices] or in desktops, the interfaces will adapt bit by bit, and the whole way we work with computers will actually change.”

**Challenges and Opportunities**

The team’s development process was not without challenges. Wolfgruber says, “We found it challenging to keep the Workbench, the maximized areas, the albums etc., synchronized with the underlying database, so that the right images are in the right position. Also, keeping the application running smoothly, even with big data, needed deeper attention.”

Through the development process, key opportunities included:
Case Study: 4tiitoo

Creating a photo-organizing application that doesn’t require the user to sit uncomfortably at a desk for a long period of time

- Developing an application that could work with a variety of operating systems
- Finding the right voice recognition software
- Determining which modalities and commands to include for the fastest and most immersive user experience
- Implementing an abstraction layer for commands
- Fine-tuning input recognition

The Ultrabook Experience Software Challenge

In developing for Ultrabook devices, the 4tiitoo team was most impressed with their touch capabilities and sensors as well as the thin design. As first place winners of the Ultrabook Experience Software Challenge, the team clearly made good use of these features.

For the 2012 Ultrabook Experience Software Challenge, EMEA-based software developers and independent software vendors were invited to submit their creative ideas for software applications that take advantage of the latest Ultrabook functionality, including touch, gesture, and voice recognition. The objective was to foster innovation and developer creativity for a more immersive and intuitive user experience with Ultrabook devices. Thirty participants were selected, with nominees from 11 different countries: the United Kingdom, Spain, Italy, Germany, the Netherlands, Russia, Romania, Israel, France, Greece, and Malta. Each participant received an Ultrabook Software Development Platform and six weeks to finish the application. The jury consisted of engineering, marketing, and retail representatives within Intel.

In terms of next steps, the team hopes to enhance NUIA Imagine with additional basic editing tools, deeper integration of speech control (e.g., voice tagging of pictures), and integration of social media and cloud functionalities.

Summary

The development company 4tiitoo was selected to participate (and ultimately won first place) in the Ultrabook Experience Software Challenge. For the challenge, the company developed NUIA Imagine, an application that helps users organize photos into albums. Users can provide input to the application with keyboard and mouse, touch, voice recognition, and eye tracking, enabling them to choose the most comfortable and natural way to interact with the software. The team decided to make NUIA Imagine a desktop application so that it would be available for use with as many operating systems as possible. The team used the Qt framework and the NUIA SDK to program the application. The types of input and the commands available in the application were based on those that would be most intuitive for typical users based on previous experiences with smartphones, tablets, and other software. Eye tracking is the newest technology used in the
application. The most challenging part of the development process was keeping the app running smoothly even with big data.

**Company**

4tiitoo AG is a pioneer in developing software solutions focused on natural user experience and business models for next-generation computing devices. The company was founded in 2007 to bring a more intuitive and natural user experience to daily computer interaction.

With a focus on touch at the time, 4tiitoo launched the tablet PC, WeTab, in 2010. Since then, the company has extended development to a multisensor user experience and provides intuitive software solutions across platforms, sensors, and languages.

4tiitoo’s latest product, the [NUIA (Natural User Interaction) Software Suite](#), offers original equipment manufacturers and sensor vendors a high-level abstraction layer with an extension model that easily enables existing applications for new computing capabilities. For developers, the NUIA technology provides a simple way to create applications based on the comprehensive NUIA SDK.

**About the Author**

Karen Marcus, M.A., is an award-winning technology marketing writer with 16 years of experience. She has developed case studies, brochures, white papers, data sheets, solution briefs, articles, website copy, video scripts, and other documents for such companies as Intel, IBM, Samsung, HP, Amazon Web Services, Microsoft, and EMC. Karen is familiar with a variety of current technologies, including cloud computing, IT outsourcing, enterprise computing, operating systems, application development, digital signage, and personal computing.
Notices

Intel, the Intel logo, and Ultrabook are trademarks of Intel Corporation in the US and/or other countries. Copyright © 2013 Intel Corporation. All rights reserved.

*Other names and brands may be claimed as the property of others.