

# 모바일 기기의 사용자 인터페이스 기술 동향에 관한 연구

조신규\*

\*한양대학교 정보통신컴퓨터대학원  
e-mail : [inficity@gmail.com](mailto:inficity@gmail.com)

## A Trend Research on User Interface Technology for Mobile Device

Sin-gyu Jo\*

\*Dept. of Computer Science, Hanyang University

### 요 약

최근 향상된 사용자 경험(User Experience)을 기반으로 한 제품들이 많은 성공을 거두게 되면서 휴대폰부터 MP3 등의 모바일 기기까지 넓은 영역으로 향상된 사용자 인터페이스를 도입하고자 하는 연구가 많이 진행되고 있다. 본 논문에서는 모바일 기기의 사용자 인터페이스에 관련한 하드웨어 및 소프트웨어의 기술 동향에 관해 알아본다.

### 1. 서론

모바일 기기들은 기기의 특성상 제품의 크기가 일반 가전기기에 비해서 작은 경우가 많다. 작은 크기에서 다양한 기능들을 작동시켜야 하기 때문에 다양한 방식으로 사용자 인터페이스가 고안되어왔다. 최근에는 터치 스크린을 활용하여 입/출력 인터페이스를 하나로 통일시키고 사용자가 보면서 만지고 움직이는 형태의 사용자 인터페이스가 많이 활용되고 있다. 과거에는 모바일 기기에 사용되는 하드웨어의 성능이 제품 본연의 기능을 겨우 수행할 정도이었으나, 최근 비약적인 성능 향상을 통해 MP3 플레이어, PMP, 전자사전, 네비게이션 등의 제품들이 모두 통합되는 컨버전스의 시대를 맞이하였다.

이렇게 하나의 제품이 많은 기능이 포함되면서 조작의 복잡도도 함께 상승하였고 이를 편리하게 조작하기 위한 사용자 인터페이스의 품질이 제품의 품질을 판단하게 하는 기준이 되었다[4].

이런 사용자 인터페이스가 발전하기 위해서는 하드웨어의 발전과 이를 지원할 수 있는 플랫폼이 있어야 한다. 본 논문에서는 사용자 인터페이스의 동향을 알아보고, 이를 지원하기 위한 플랫폼과 하드웨어의 발전 동향에 대해서 알아본다. 그리고, 모바일 기기의

사용자 인터페이스 개발을 위한 방향에 대해 제안한다.

### 2. 사용자 인터페이스 동향

#### 2.1. PC OS의 사용자 인터페이스

사람의 감각 기관중에 가장 많이 사용되는 것이 눈이기 때문에 그래픽적인 요소가 가장 먼저 발전하고 있다. 이런 면에서 GUI가 가장 빠르게 발전하고 있는 것은 PC 어플리케이션들의 사용자 인터페이스이다.



(그림 1) PC OS의 사용자 인터페이스 화면

#### 2.1.1. Windows Vista

이전 Windows XP에서는 사용자 인터페이스 구현을 위한 기술들이 표 1과 같이 여러가지로 분리되어 있

었다. 이를 WPF(Windows Presentation Foundation)을 통해 하나의 기술 범위 안에서 모두 구현가능하도록 하였다. Aero UI 기술을 통해 작업중인 윈도우들을 입체공간상에 펼쳐놓고 볼 수 있는 기능을 제공한다 [3].

구분	기능
WindowsForms	GUI(폼 및 버튼)
PDF	고정된 형식의 문서
WindowsForms/GDI+	이미지 제어, 2 차원 그래픽
WindowsMediaPlayer	비디오 및 오디오
Direct3D	3 차원 그래픽

<표 1> Windows XP 에서 UI구현을 위한 기술



(그림 3) Apple iPhone, LG Arena phone

2.1.2. Mac OS X

OS 의 그래픽 레이어에서 3D, 벡터, 동영상의 혼용을 지원하는 Core Animation API 를 제공한다. 이를 통해 응용프로그램에서 멀티미디어 혼용 효과를 제공할 수 있다.

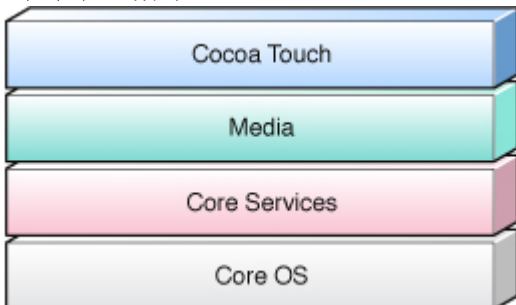
2.2. 모바일 기기의 사용자 인터페이스

2.2.1. Apple iPhone

입력 인터페이스는 정전압식 멀티 터치 스크린과 가속도 센서를 사용하였다. 윈터치 스크린과 달리 Pinch zoom 기능을 지원할 수 있어서 작은 화면에서도 빠르게 zoom in/out 을 할 수 있다.

아이폰의 OS 는 Mac OS X 를 기반으로 아이폰을 위해 만들어진 OS 이다. Mac OS X 와 마찬가지로 Core Animation API 를 제공하며, 멀티 터치 인터페이스를 위한 Cocoa Touch 를 제공한다.

아이폰은 어플리케이션 제작 후 이를 배포하기 위해서는 App Store 라는 자체적인 온라인 마켓을 통해 하 하는데, UI Guideline 을 정해놓고 이를 준하는 소프트웨어만 배포를 허락한다. 이를 통해 대부분의 아이폰 프로그램은 인터페이스 및 제어방식이 통일성을 유지하고 있다.



(그림 2) iPhone OS technology layers[1]

2.2.2. SKY IM-S410

마이크 및 카메라를 통해 사람의 행동을 인식하는 입력 인터페이스를 적용하였다. 마이크를 향해 입김을 불면 소리의 크기를 인식해서 각종 액션을 취하는 방식과 카메라로 사물이나 사람을 인식하여 모션을 인식하는 방식을 사용했다.

2.2.3. LG Arena

메뉴 UI 를 정육면체 큐브에 배치했다. 기존 핸드폰들이 아이콘이나 이미지의 가공을 통해 2D 에서 3D 의 느낌이 나게끔 효과를 준 반면 실제 3D 그래픽을 적용하여 메뉴 UI 를 구성했다.

3. 플랫폼

모바일 기기의 하드웨어 성능이 점점 발전하면서 OpenGL ES 등의 표준을 지원하는 범위가 넓어져 가고 있고, 다양한 기능 제공을 위해 특정 기기를 위해 고안된 OS 가 아닌 GPOS(General Purpose O/S)의 사용이 늘어나고 있는 추세이다.

3.1. Apple Core Animation Framework

OpenGL 을 기반으로 동영상(QuickTime), 이미지(Core Image) 벡터 그래픽(Quartz) 등을 모두 H/W 가속으로 처리하는 UI Framework 이다. 이로서 멀티미디어 UI 를 자유롭게 구성할 수 있게끔 되어있다.

3.2. Windows Mobile 7

멀티 터치를 지원하고, 터치 제스처 인식 기능을 지원하여 이를 통한 제어를 지원한다.

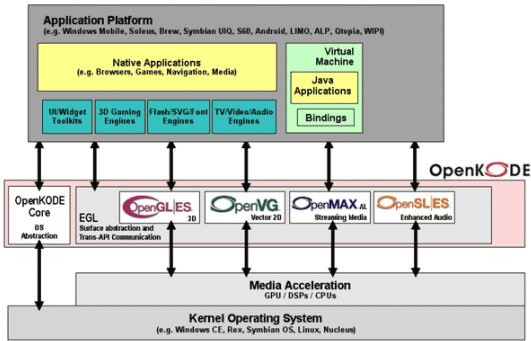


(그림 4) Features of Windows Mobile 7

3.3. Khronos OpenKODE

기존 Khronos 규격은 멀티미디어에 대한 개별 표준 API 를 지원해 왔다. 최근 멀티 플랫폼에 대한 필요성이 대두되어 OpenKODE 규격화 작업이 진행되고 있

다[2]. 제한된 모바일 기기 환경에서 미디어 혼용을 지원하기 위한 규격을 잡고 있는 중이다.



(그림 5) The OpenKODE set of native APIs

Interface Framework 를 마련하여 향상된 사용자 인터페이스를 개발을 위한 준비를 해야 될 것으로 보인다.

6. 결론

하드웨어의 발전과 플랫폼의 표준화로 인해 사용자 인터페이스의 개발 과정도 일정한 표준 형태를 띠게 될 것이고 따라서 범용 UI 프레임워크가 개발 될 수 있다. 앞으로 생길 다양한 입/출력 인터페이스를 지원하기 위한 확장성을 고려한 UI 프레임워크에 대한 연구가 필요하다.

4. Chip 및 단말기

OpenGL/ES 를 지원하는 미디어칩이 늘어남에 따라 GUI 를 순수 소프트웨어 렌더링으로 처리하던 방식에서 하드웨어 가속을 지원받을 수 있게 되었다. 이에 따라, GUI 의 반응 속도 및 애니메이션 효과를 부드럽게 처리 할수 있게 되어 향상된 UI 가 모바일 기기에 적용되게 된 직접적인 계기가 되었다.

참고문헌

- [1] iPhone Dev Center, <http://developer.apple.com/iphone/gettingstarted/docs/iphonesooverview.action>
- [2] OpenKODE, <http://www.khronos.org/openkode/>
- [3] Windows Area Technology, <http://www.microsoft.com/windows/windows-vista/features/aero.aspx>
- [4] 유재욱, “사용자 중심적 인터페이스 설계에 관한 연구”, 덕성여자대학교 2001

	OpenGL/ES	OpenVG	OpenMAX	OpenKODE
MSM7K	1.0+Ext	1.0	-	-
MSM8K	2.0	1.1	-	-
Mali200	2.0	1.0	1.0	-
OMAP3	2.0	1.0	-	준비중
APX2500	2.0	1.1	1.0	1.0
A290	2.0	1.1	-	-
MDIRAC	2.0	1.0	1.0	-

<표 2> 칩 별 표준 규격 지원 동향

	OpenGL/ES	OS	Perf
Nokia 6290	1.0/1.1	Symbian 9.x S60	240*320 106fps
Nokia N93	1.1	Symbian 9.x S60	240*320 509fps
Mototola RIZR Z8	1.1	Windows Mobile	240*320 396fps
Apple iPhone	1.1	iPhone OS	320*480 527fps
Never land	2.0	Linux	800*480 500fps

<표 3> 단말기 별 제원

5. 사용자 인터페이스 개발 방향

앞에서 살펴본 바에 따르면 최근에 모바일 기기들의 GPOS 채택 사례가 늘고 있다. 이에 대한 대응이 필요하며 향상된 사용자 인터페이스를 위해 H/W 가속 기능을 활용할 필요가 있다.

이를 위해 표준 OpenKODE 를 기반으로 한 User