



(19) 대한민국특허청(KR)  
(12) 등록특허공보(B1)

(45) 공고일자 2012년03월15일  
(11) 등록번호 10-1117859  
(24) 등록일자 2012년02월10일

(51) 국제특허분류(Int. Cl.)

G06F 3/14 (2006.01)

(21) 출원번호 10-2005-0016743

(22) 출원일자 2005년02월28일

심사청구일자 2010년02월18일

(65) 공개번호 10-2006-0043268

(43) 공개일자 2006년05월15일

(30) 우선권주장

10/789,603 2004년02월27일 미국(US)

(56) 선행기술조사문헌

JP2002268867 A\*

US06044408 A1\*

\*는 심사관에 의하여 인용된 문헌

(73) 특허권자

마이크로소프트 코포레이션

미국 워싱턴주 (우편번호 : 98052) 레드몬드 원  
마이크로소프트 웨이

(72) 발명자

쥬 치 에이치.

미국 98052 워싱턴주 레드몬드 원 마이크로소프트  
웨이 마이크로소프트 코포레이션 내

폴러 제이슨 더블유

미국 98052 워싱턴주 레드몬드 원 마이크로소프트  
웨이 마이크로소프트 코포레이션 내

(뒷면에 계속)

(74) 대리인

제일특허법인

전체 청구항 수 : 총 7 항

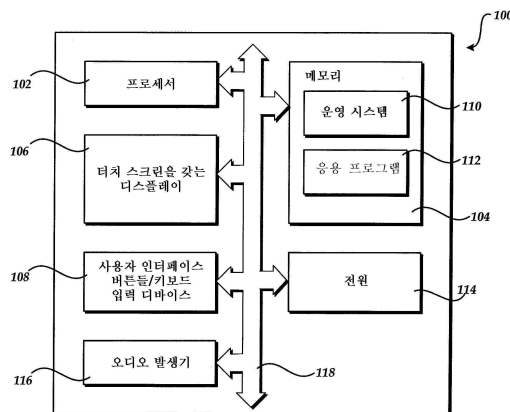
심사관 : 한선경

(54) 발명의 명칭 개선된 픽셀 밀도를 갖는 디스플레이 디바이스에서 응용 프로그램의 호환성을 인에이블하는 방법 및 장치

(57) 요약

증가된 픽셀 밀도를 갖는 디스플레이 디바이스에서 응용 프로그램의 호환성을 인에이블하기 위한 방법 및 장치가 제공된다. 일 방법에 따르면, 한 개 이상의 파라미터들을 포함하는 콜은 응용 프로그램 프로그래밍 인터페이스로 전달되어 응용 프로그램으로부터 수신되는 스크린 입력이나 출력 기능을 수행한다. 콜의 수신에 응답하여, 응용 프로그램이 저 픽셀 밀도를 갖는 디스플레이 디바이스 또는 고 픽셀 밀도를 갖는 디스플레이 디바이스에서 사용하도록 구성되었는지에 대해 판정이 된다. 응용 프로그램이 저 픽셀 밀도를 갖는 디스플레이 디바이스에서 사용하도록 구성되었다는 판정에 응답하여, 파라미터들은 고 픽셀 밀도 디스플레이 디바이스를 위해 스케일되고, 응용 프로그램 프로그래밍 인터페이스는 스케일된 파라미터들을 사용하여 콜된다.

대표도 - 도1



(72) 발명자

**벤틀리 키이스 씨.**

미국 98052 워싱턴주 레드몬드 원 마이크로소프트  
웨이 마이크로소프트 코포레이션 내

**게리 론 오.**

미국 98052 워싱턴주 레드몬드 원 마이크로소프트  
웨이 마이크로소프트 코포레이션 내

**코크 지크**

미국 98052 워싱턴주 레드몬드 원 마이크로소프트  
웨이 마이크로소프트 코포레이션 내

## 특허청구의 범위

### 청구항 1

저 픽셀 밀도를 갖는 디스플레이 디바이스와 함께 사용하도록 구성된 응용 프로그램이 고 픽셀 밀도를 갖는 디스플레이 디바이스를 이용할 수 있게 하는 방법으로서,

상기 응용 프로그램과 응용 프로그램 프로그래밍 인터페이스(API) 사이에 연결된 양방향 변환 레이어에서, 스크린 입력 또는 출력 기능을 수행하기 위한 상기 응용 프로그램 프로그래밍 인터페이스로 향하는 콜(call)을 상기 응용 프로그램으로부터 수신하는 단계 - 상기 콜은 하나 이상의 파라미터들을 포함함 -;

상기 콜의 수신에 응답하여, 상기 응용 프로그램이 저 픽셀 밀도를 갖는 디스플레이 디바이스와 함께 사용하도록 구성되어 있는지 또는 고 픽셀 밀도를 갖는 디스플레이 디바이스와 함께 사용하도록 구성되어 있는지를 판정하는 단계 - 상기 응용 프로그램이 해당 디스플레이 디바이스와 함께 사용하도록 구성되어 있는지를 판정하는 단계는, 상기 응용 프로그램의 특성을 조사하여, 상기 응용 프로그램을 컴파일하는 소프트웨어 개발 키트의 버전 번호를 결정하는 단계 - 상기 버전 번호는 상기 응용 프로그램의 컴파일 시에 상기 응용 프로그램 내에 저장되며, 상기 버전 번호는 상기 응용 프로그램의 픽셀 밀도 지원 레벨에 대응함 -, 상기 버전 번호가 미리 정해진 값보다 크거나 상기 미리 정해진 값과 동등한 경우에는, 상기 응용 프로그램이 상기 고 픽셀 밀도를 갖는 디스플레이 디바이스와 함께 사용하도록 구성되어 있는 것으로 판정하는 단계, 및 상기 버전 번호가 상기 미리 정해진 값보다 작은 경우에는, 상기 응용 프로그램이 상기 저 픽셀 밀도를 갖는 디스플레이 디바이스와 함께 사용하도록 구성되어 있는 것으로 판정하는 단계를 포함함 -;

상기 응용 프로그램이 상기 저 픽셀 밀도를 갖는 디스플레이 디바이스와 함께 사용하도록 구성되어 있는 것으로 판정한 것에 응답하여, 상기 양방향 변환 레이어에서, 상기 고 픽셀 밀도의 디스플레이 디바이스의 픽셀 밀도를 상기 저 픽셀 밀도의 디스플레이 디바이스의 픽셀 밀도로 나눈 비율을 상기 파라미터들에 승산함으로써 상기 파라미터들을 스케일(scale)하며, 상기 스케일된 파라미터들로 상기 응용 프로그램 프로그래밍 인터페이스를 콜하는 단계;

상기 응용 프로그램 프로그래밍 인터페이스가 적어도 하나의 픽셀 값을 리턴하였는지를 판정하는 단계; 및

상기 응용 프로그램 프로그래밍 인터페이스가 적어도 하나의 픽셀 값을 리턴한 것으로 판정한 것에 응답하여, 상기 적어도 하나의 픽셀 값을 상기 저 픽셀 밀도의 디스플레이 디바이스의 픽셀 밀도로 스케일하며, 상기 스케일된 픽셀값을 상기 응용 프로그램에 리턴하는 단계를 포함하는 방법.

### 청구항 2

제1항에 있어서, 상기 저 픽셀 밀도의 디스플레이 디바이스는 인치당 96 도트의 픽셀 밀도를 갖는 방법.

### 청구항 3

제2항에 있어서, 상기 고 픽셀 밀도의 디스플레이 디바이스는 인치당 192 도트의 픽셀 밀도를 갖는 방법.

### 청구항 4

컴퓨터에 의해 실행되는 경우, 상기 컴퓨터가 제1항의 방법을 수행하도록 하는 컴퓨터 실행가능한 명령어들이 저장된 컴퓨터 판독가능 저장 매체.

### 청구항 5

저 픽셀 밀도를 갖는 디스플레이 디바이스와 함께 사용하도록 생성된 응용 프로그램이 고 픽셀 밀도를 갖는 디스플레이 디바이스를 이용할 수 있게 하도록 구성되는 컴퓨터 시스템으로서,

중앙 프로세싱 유닛;

고 픽셀 밀도를 갖는 디스플레이 디바이스; 및

메모리

를 포함하며,

상기 메모리는,

상기 중앙 프로세싱 유닛 상에서 실행하기 위한 운영 시스템,

저 픽셀 밀도를 갖는 디스플레이 디바이스와 함께 사용하도록 생성되며, 상기 운영 시스템 상에서 실행하기 위한 응용 프로그램,

디스플레이 디바이스에 대해 입력 및 출력 동작들을 수행하기 위한 응용 프로그램 프로그래밍 인터페이스, 및

상기 응용 프로그램에 의한 상기 응용 프로그램 프로그래밍 인터페이스로의 콜들을 인터셉트하고 - 상기 콜들은 파라미터들을 포함함 -, 디스플레이 디바이스에 대해 상기 파라미터들을 스케일하며, 상기 스케일된 파라미터들로 상기 응용 프로그램 프로그래밍 인터페이스를 콜하기 위한 변환 레이어

를 저장하도록 동작하고,

상기 변환 레이어는, 상기 응용 프로그램이 저 픽셀 밀도를 갖는 디스플레이 디바이스와 함께 사용하도록 구성되어 있는지 또는 고 픽셀 밀도를 갖는 디스플레이 디바이스와 함께 사용하도록 구성되어 있는지를 판정하며, 상기 응용 프로그램이 상기 저 픽셀 밀도를 갖는 디스플레이 디바이스와 함께 사용하도록 구성되어 있는 것으로 판정한 것에 응답하여, 상기 고 픽셀 밀도의 디스플레이 디바이스의 픽셀 밀도를 상기 저 픽셀 밀도의 디스플레이 디바이스의 픽셀 밀도로 나눈 비율을 상기 파라미터들에 승산함으로써 상기 파라미터들을 스케일하기 위해서,

상기 응용 프로그램의 특성을 조사하여, 상기 응용 프로그램을 컴파일하는 소프트웨어 개발 키트의 버전 번호를 결정하고 - 상기 버전 번호는 상기 응용 프로그램의 컴파일 시에 상기 응용 프로그램 내에 저장되며, 상기 버전 번호는 상기 응용 프로그램의 픽셀 밀도 지원 레벨에 대응함 -,

상기 버전 번호가 미리 정해진 값보다 크거나 상기 미리 정해진 값과 동등한 경우에는, 상기 응용 프로그램이 상기 고 픽셀 밀도를 갖는 디스플레이 디바이스와 함께 사용하도록 구성되어 있는 것으로 판정하고,

상기 버전 번호가 상기 미리 정해진 값보다 작은 경우에는, 상기 응용 프로그램이 상기 저 픽셀 밀도를 갖는 디스플레이 디바이스와 함께 사용하도록 구성되어 있는 것으로 판정하고,

상기 응용 프로그램 프로그래밍 인터페이스가 적어도 하나의 픽셀 값을 리턴하였는지를 판정하고,

상기 응용 프로그램 프로그래밍 인터페이스가 적어도 하나의 픽셀 값을 리턴한 것으로 판정한 것에 응답하여, 상기 적어도 하나의 픽셀 값을 상기 저 픽셀 밀도의 디스플레이 디바이스의 픽셀 밀도로 스케일하며, 상기 스케일된 픽셀값을 상기 응용 프로그램에 리턴하도록 또한 동작하는 컴퓨터 시스템.

## 청구항 6

제5항에 있어서, 상기 변환 레이어는, 상기 응용 프로그램이 상기 고 픽셀 밀도를 갖는 디스플레이 디바이스와 함께 사용하도록 구성되어 있는 것으로 판정한 것에 응답하여, 상기 파라미터들을 스케일하지 않고 상기 응용 프로그램 프로그래밍 인터페이스를 콜하도록 또한 동작하는 컴퓨터 시스템.

## 청구항 7

제6항에 있어서, 상기 고 픽셀 밀도를 갖는 디스플레이 디바이스는 192 dpi의 픽셀 밀도를 가지며, 상기 저 픽셀 밀도를 갖는 디스플레이 디바이스와 함께 사용하도록 생성된 상기 응용 프로그램은 96 dpi의 픽셀 밀도를 갖는 디스플레이 디바이스와 함께 사용하도록 구성되는 컴퓨터 시스템.

## 청구항 8

삭제

## 청구항 9

삭제

## 청구항 10

삭제

청구항 11

삭제

청구항 12

삭제

청구항 13

삭제

청구항 14

삭제

청구항 15

삭제

청구항 16

삭제

청구항 17

삭제

청구항 18

삭제

청구항 19

삭제

명세서

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

- [0015] 본 발명은 일반적으로 응용 프로그램들에 대한 백워드 호환성의 분야에 관한 것이고, 더 구체적으로는 개선된 픽셀 밀도를 갖는 디스플레이 디바이스에서 응용 프로그램의 백워드(backward) 호환성을 인에이블하는 분야에 관한 것이다.
- [0016] 핸드헬드 컴퓨팅 디바이스들(또한 본 명세서에서 개인 정보 단말기("PDA")로 언급됨)이 진화하면서, 이 클래스의 디바이스들의 컴퓨팅 능력들이 크게 증가해 왔다. 특히 크게 개선되어온 PDA의 한 개의 하드웨어 컴포넌트는 디스플레이 스크린이다. 시간이 지남에 따라, PDA 디스플레이 스크린의 해상도와 픽셀 밀도 모두는 크게 증가해 왔다. 본 명세서에 사용되는 바와 같이, "해상도"라는 용어는 디스플레이 스크린 상에서 액세스할 수 있는 수평과 수직 픽셀들의 수를 언급한다. 예를 들어, 일부 PDA들은 240 수평 픽셀과 320 수직 픽셀을 갖는 디스플레이 스크린들을 사용하고, 한편 다른 PDA들은 480 수평 픽셀과 640 수직 픽셀을 갖는 디스플레이 스크린들을 사용한다. 본 명세서에서 사용되는 바와 같이, "픽셀 밀도"라는 용어는 측정의 명시된 유닛 내의 디스플레이 스크린 상에서 액세스할 수 있는 픽셀들의 수를 언급한다. 통상적으로, 픽셀 밀도는 인치 당 도트("dpi")들 (또는 픽셀들)을 말한다. 예를 들어, 일부 PDA들은 96dpi를 갖는 디스플레이 스크린들을 사용하고, 한편 다른 PDA들은 192dpi의 개선된 픽셀 밀도를 갖는 디스플레이 스크린을 사용한다.
- [0017] 현대 PDA들의 개선된 해상도와 픽셀 밀도는 사용자들에게는 이익이다. 개선된 해상도와 픽셀 밀도는 더 많은 정보가 스크린에 디스플레이되도록 하여, 문서와 웹 페이지의 손쉬운 뷰잉 및 다른 기능들을 수행하도록

한다. 개선된 해상도는 또한 명료성과 관독성을 향상시킨다. 그러나, 개선된 해상도와 픽셀 밀도는 PDA를 위한 프로그램을 작성하는 프로그래머들에게는 난제(challenge)로 되어왔다. 이들 난제들은 일반적으로 디스플레이 스크린이 특정 픽셀 밀도를 갖는다는 가정을 하는 다수의 PDA 응용 프로그램들이 씌어졌다는 사실에 의해 유발되어 왔다. 응용 프로그램들은 스크린 관련 기능들을 수행하기 위해 콜하는 운영 시스템이 제공하는 응용 프로그램 프로그래밍 인터페이스("APIs")는 종종 픽셀들을 파라미터들로서 택한다는 사실에 의해 이 문제가 구성된다. 그러므로, 예를 들어, 9dpi 디스플레이 스크린과 사용하기 위해 구성된 응용 프로그램이 1/2인치 너비의 온-스크린 사용자 인터페이스 버튼을 그리면, 그것은 48 픽셀 너비의 버튼을 생성하기 위해 API를 콜할 것이다. 그러나, 동일 응용 프로그램이, 예를 들어, 192dpi의 개선된 픽셀 밀도를 갖는 PDA 상에서 실행되면, 동일 48픽셀 너비 버튼은 단지 1/4인치 너비일 것이다. 개선된 픽셀 밀도를 갖는 디바이스들의 등장 이전에 생성된 응용 프로그램들은 증가된 밀도를 인식 못하기 때문에, 이들 프로그램들에 의해 그려진 모든 사용자 인터페이스 객체들은 의도된 바보다 더 작게 디스플레이될 것이다. 스크린이 터치 민감형 디바이스로서 사용되는 입력 동작들은 또한 그런 레거시(legacy) 응용 프로그램들에 의해 이해될 수 있다.

[0018] 이 문제의 한 가지 해결책은 응용 프로그램 프로그래머들이 그들의 응용 프로그램들을 재프로그래밍하여 더 새로운 PDA들의 개선된 픽셀 밀도를 지원하도록 요구하는 것이다. 그러나, 이것은 실제로 응용 프로그램을 재코딩해야 하는 프로그래머에게는 성가신 일이고 시간소비적일 것이다. 이것은 또한 응용 프로그램이 재코드되어 그 응용 프로그램을 증가된 픽셀 밀도를 갖는 새 디바이스에서 함께 사용할 수 있을 때까지 기다려야 하는 사용자들에게도 성가신 일일 것이다. 일부 경우들에서, 프로그래머들은 고 밀도 디스플레이들을 위한 응용 프로그램을 재코드하지 않도록 단순히 결정할 것이다. 그 다음, 그 프로그램의 사용자들은 고 밀도의 디스플레이 이 스크린의 장점을 취하기 위해 다른 프로그램으로 이동하도록 강제될 것이다.

[0019] 이 문제의 다른 해결책은 고 해상도 스크린들 상의 스크린 관련 기능들을 수행하기 위해 API들의 새로운 세트를 생성하는 것이다. 그러나, 이 해결책은 또한, 그것은 레거시 응용 프로그램들이 고 밀도 디스플레이들을 사용하도록 허용하지 않고, 또한 새로운 고 밀도-인지 응용 프로그램들을 생성하는 프로그래머들이 완전히 새로운 API를 사용할 것을 요구하므로 바람직하지 않다. 그러므로, 레거시 응용 프로그램들이 응용 프로그램들을 재작성(rewrite)하지 않고 개선된 픽셀 밀도 스크린에 정확하게 입력하고 출력할 수 있고, 새 고 밀도-인지 응용 프로그램들이 레거시 응용 프로그램들처럼 스크린 동작들을 위해 동일 세트의 API들을 사용하도록 하고, 스크린 입력과 출력을 수행하는 API들을 다시 작성하지 않고 이 기능을 인에이블하기 위해 바람직하다. 그것은 본 발명의 다양한 실시예들이 만들어지는 이들 고려사항들과 다른 것들에 대한 것이다.

### 발명이 이루고자 하는 기술적 과제

[0020] 본 발명에 따르면, 상술된 그리고 다른 문제들은 증가된 픽셀 밀도를 갖는 디스플레이 디바이스들에서 응용 프로그램의 호환성을 인에이블하기 위한 방법 및 장치에 의해 해결된다. 본 발명에 의해 제공된 기능을 사용하여, 레거시 응용 프로그램들은 수정없이 개선된 픽셀 밀도 스크린으로 정확하게 입력하고 출력할 수 있다. 더욱이, 고 밀도 디스플레이를 인지하는 응용 프로그램들은 개선된 픽셀 밀도를 인지하지 못하는 레거시 응용 프로그램들처럼 스크린 동작들을 하기 위해 API들의 동일 세트를 사용할 수 있다. 이 기능은 재작성되는 스크린 입력과 출력을 수행하는 API들을 요구하지 않고 제공될 수 있다.

[0021] 본 발명의 일 양태에 따라, 저 픽셀 밀도를 갖는 디스플레이 디바이스에서 사용하도록 구성된 응용 프로그램이 고 픽셀 밀도를 갖는 디스플레이 디바이스를 완전히 사용하도록 하기 위해 제공된다. 그 방법에 따르면, PDA 나 기타 유형의 컴퓨팅 디바이스 상에서 실행하는 응용 프로그램은 API로의 요구(request)나 콜(call)을 발생시켜, 스크린에 그리는 것이나 터치 민감형 스크린으로부터 입력을 수신하는 것과 같은, 스크린 입력이나 출력 기능을 수행할 것이다. API 콜은 수행되는 동작을 정의하는 한 개 이상의 파라미터들을 포함할 수 있고, 픽셀 유닛들을 사용하여 표현될 수 있다. 예를 들어, API로 콜이 되어 48 픽셀 너비의 온-스크린 사용자 인터페이스 버튼을 그릴 수 있다.

[0022] 그런 콜이 수신될 때, 콜은 의도된 API에 도달되기 전에 인터셉트(intercept)될 수 있다. 그 다음, 응용 프로그램이 저 픽셀 밀도를 갖는 디스플레이 디바이스 또는 고 픽셀 밀도를 갖는 디스플레이 디바이스에서 사용하도록 구성되는지에 대해 판정될 수 있다. 이 판정은 응용 프로그램이 컴파일되는 소프트웨어 개발 키트 버전 번호를 결정하는 응용 프로그램의 특성을 조사하여 될 수 있다. 소정의 번호보다 더 크거나 같은 버전 번호들을 갖는 응용 프로그램들은 고 픽셀 밀도를 갖는 디스플레이들을 지원한다고 가정된다. 소정의 번호보다 더 낮은 버전 번호들을 갖는 응용 프로그램들은 저 픽셀 밀도를 갖는 디스플레이들만을 지원한다고 가정된다.

- [0023] 응용 프로그램이 저 픽셀 밀도를 갖는 디스플레이 디바이스에서 사용하도록 구성되면, API로 수신되는 파라미터들은 고 픽셀 밀도 디스플레이 디바이스의 픽셀 밀도로 스케일된다. 예를 들어, 응용 프로그램이 96dpi 픽셀 밀도를 위해 구성되고, 고 픽셀 밀도를 갖는 디스플레이 디바이스는 192dpi의 픽셀 밀도를 가지면, 파라미터들은 두배가 된다. 그 다음, 의도된 응용 프로그램 프로그래밍 인터페이스로 스케일된 파라미터들을 갖는 콜이 되어진다. 응용 프로그램이 고 픽셀 밀도를 갖는 디스플레이 디바이스에서 사용하도록 구성되면, API 콜은 API 함수 콜이 파라미터들을 스케일하지 않고 API로 직접 전달된다. 이 방식으로, 고 픽셀 밀도의 디스플레이 스크린과의 사용을 위해 구성된 응용 프로그램들은 의도된대로 실행할 것이다.
- [0024] 예를 들어, API가 터치 민감형 디스플레이로부터 스타일러스 입력을 수신하기 위해 사용될 때와 같이, 파라미터들이 API 콜로부터 리턴되면, 이들 파라미터들은 저 픽셀 밀도로 스케일된다. 스크린 I/O 동작들을 수행하는 네이티브(native) API들로의 함수 콜들을 인터셉트하고, 파라미터들을 스케일하고, 그 다음에 그 스케일된 파라미터들로 네이티브 API들을 콜하여, 고 픽셀 밀도의 디스플레이 스크린들을 위한 지원을 포함하지 않는 레거시 응용 프로그램들은 응용 프로그램들이나 API들의 제작성을 요구하지 않고 고 해상도 디바이스의 전체 디스플레이 스크린을 사용할 수 있다.
- [0025] 본 발명의 다른 양태에 따르면, 저 픽셀 밀도를 갖는 디스플레이 디바이스에서 사용하도록 생성된 응용 프로그램이 고 픽셀 밀도를 갖는 디스플레이 디바이스를 완전히 사용하도록 구성되는 컴퓨터 시스템이 제공된다. 컴퓨터 시스템은 중앙 프로세싱 유닛("CPU"), 고 픽셀 밀도를 갖는 디스플레이 디바이스, 및 한 개 이상의 실행 가능한 프로그램들을 저장하기 위해 동작하는 메모리를 포함한다. 특히, 메모리는 CPU 상에서의 실행을 위한 운영 시스템, 저 픽셀 밀도를 갖는 디스플레이 디바이스에서 사용하도록 구성된 응용 프로그램, 디스플레이 디바이스 상에서 I/O 동작들을 수행하는 API, 및 변환 레이어(translation layer)를 저장하기 위해 동작한다.
- [0026] 변환 레이어는 응용 프로그램에 의해 API로 발생된 콜들을 인터셉트하기 위해 동작한다. 그런 콜들이 수신될 때, 변환 레이어는 콜과 함께 수신된 임의의 파라미터들을 설치된 디스플레이 스크린의 고 픽셀 밀도로 스케일한다. 그 다음, 변환 레이어는 스케일된 파라미터들로 의도된 API를 콜한다. 예를 들어, API가 터치 민감형 디스플레이로부터 입력을 수신하기 위해 사용될 때와 같이, 파라미터들이 API 콜로부터 리턴되면, 변환 레이어는 파라미터 값들을 응용 프로그램에 의해 사용되는 저 픽셀 밀도로 스케일한다.
- [0027] 변환 레이어는 또한 콜하는 응용 프로그램이 저 픽셀 밀도를 갖는 디스플레이 디바이스에서 사용 또는 고 픽셀 밀도를 갖는 디스플레이 디바이스에서 사용하도록 구성되는지를 판정하기 위해 동작할 것이다. 응용 프로그램이 저 픽셀 밀도 디스플레이와 사용하기 위해 구성되면, 변환 레이어는 상술된 방식으로 파라미터들을 스케일한다. 그러나, 응용 프로그램이 고 픽셀 밀도를 갖는 디스플레이와의 사용을 위해 생성되었다고 변환 레이어가 판정하면, 변환 레이어는 파라미터들을 스케일하지 않고 의도된 API를 콜한다. 콜하는 응용 프로그램이 저 픽셀 또는 고 픽셀 밀도를 지원하는지에 대한 판정은 응용 프로그램이 컴파일되는 소프트웨어 개발 키트의 버전 번호를 지시하는 응용 프로그램의 특성을 조사하여 이루어질 수 있다.
- [0028] 본 발명은 컴퓨터 프로세스, 컴퓨팅 시스템, 또는 컴퓨터 프로그램 제품이나 컴퓨터 판독가능 매체와 같은 제품으로서 구현될 수 있다. 컴퓨터 프로그램 제품은 컴퓨터 시스템에 의해 판독가능한 컴퓨터 저장 매체일 수 있고, 명령어들의 컴퓨터 프로그램을 인코딩하여 컴퓨터 프로세스를 실행할 수 있다. 컴퓨터 프로그램 제품은 또한 컴퓨팅 시스템에 의해 판독가능한 반송파의 전파된 신호일 수 있고, 명령어들의 컴퓨터 프로그램을 인코딩하여 컴퓨터 프로세스를 실행할 수 있다.
- [0029] 본 발명을 특성화하는 이들과 다양한 다른 특징들 및 장점들은 다음의 상세한 설명과 연관된 도면들의 검토의 관독으로부터 명백해질 것이다.

### 발명의 구성 및 작용

- [0030] 유사 부호들은 유사 소자들을 나타내는 도면들을 이제 참조하면서, 본 발명의 다양한 양태들이 기재된다. 도 1과 다음의 논의는 본 발명이 구현될 수 있는 적절한 컴퓨팅 환경의 간략하고, 일반적인 설명을 제공하려고 의도된다. 도 1의 컴퓨팅 환경은 설명적일 뿐이고, 본 발명을 제한하려는 의미는 아님을 이해해야 한다.
- [0031] 도 1은, 예를 들어, PDA와 같은, 핸드헬드 컴퓨팅 디바이스에서 발견될 수 있는 것과 같은 컴퓨팅 환경을 도시한다. 그러나, 본 발명은 PDA들에 제한되지 않음을 이해해야 한다. 본 발명은 멀티프로세서 시스템, 마이크로프로세서-기반이나 프로그램가능한 소비자 전자제품, 통신망 PC, 미니 컴퓨터, 메인프레임 컴퓨터, 무선 전화, 호출기 등을 포함하는 기타 컴퓨터 시스템 구조들과 실시될 수 있다. 본 발명은 또한 통신망을 통해 링크된 원격 프로세싱 디바이스들에 의해 작업들이 수행되는 분산 컴퓨팅 환경들에서 실시될 수 있다. 분산 컴퓨팅 환경



에서, 프로그램 모듈들은 로컬 및 원격 메모리 저장 디바이스들 모두에 위치될 수 있다.

[0032] 일반적으로, 프로그램 모듈들은 특정 작업들을 수행하거나 특정 추상 데이터 유형들을 구현하는 루틴, 프로그램, 컴포넌트, 데이터 구조, 및 기타 유형들의 구조를 포함한다. 더욱이, 당업자들이라면, 본 발명이 데스크톱 컴퓨팅 시스템, 멀티프로세서 시스템, 마이크로프로세서-기반이나 프로그램가능한 소비자 전자제품, 미니컴퓨터, 메인프레임 컴퓨터 등을 포함하는 기타 컴퓨터 시스템 구조들과 함께 실시될 수 있다. 본 발명은 또한 통신망을 통해 링크된 원격 프로세싱 디바이스들에 의해 작업들이 수행되는 분산 컴퓨팅 환경에서 실시될 수 있다. 분산 컴퓨팅 환경에서, 프로그램 모듈들은 로컬 및 원격 메모리 저장 디바이스들 모두에 위치될 수 있다.

[0033] 도 1을 참조하면, 본 발명을 구현하는 시스템의 예는 핸드헬드 컴퓨팅 디바이스(100)를 포함한다. 컴퓨팅 디바이스(100)는 프로세서(102), 메모리(104), 및 디스플레이(106)를 갖고 있고, 사용자 인터페이스 버튼들 및/또는 키보드 입력 디바이스(108)를 포함할 수 있다. 메모리(104)는 일반적으로 휘발성 메모리(예를 들어, RAM) 및 비휘발성 메모리(예를 들어, ROM, PCMCIA 카드 등)를 포함한다. 운영 시스템(110)은 메모리(104)에 상주하고, 프로세서(102) 상에서 실행한다. 핸드헬드 컴퓨팅 디바이스(100)는 마이크로소프트사의 "윈도우즈 모바일 2003" 운영 시스템, 팜소스(PALMSOURCE)로부터의 "팜(PALM) OS" 운영 시스템, 또는 핸드헬드나 데스크톱 컴퓨터를 운영하기 위해 적합한 다른 운영 시스템과 같은 운영 시스템(110)을 포함한다.

[0034] 한 개 이상의 응용 프로그램들(112)은 메모리(104)로 로드되고, 운영 시스템(110) 상에서 실행된다. 응용 프로그램들의 예는 이메일 프로그램, 스케줄링 프로그램, PIM(Personal Information Management) 프로그램, 문서 편집 프로그램, 스프레드시트 프로그램 등을 포함한다. 응용 프로그램(112)의 다른 예는 마이크로소프트사의 "포켓 인터넷 익스플로러(POCKET INTERNET EXPLORER)" 웹 브라우저 응용 프로그램 또는 다른 웹 브라우저와 같은 웹 브라우저 프로그램 모듈이다.

[0035] 핸드헬드 컴퓨팅 디바이스(100)는 한 개 이상의 전지들로서 구현된 전원(114)을 갖는다. 전원(114)은, AC 어댑터나 전원이 연결된 도킹 크래들과 같은, 빌트-인(built-in) 전지들을 오버라이드(override)하거나 재충전하는 외부 전원을 더 포함할 수 있다. 핸드헬드 컴퓨팅 디바이스(100)는 또한 오디오 출력을 발생시키는 오디오 발생기(116)를 포함하여 핸드헬드 디바이스(100) 내에 포함된 스피커에서 플레이백될 수 있도록 할 수 있다. 핸드헬드 컴퓨팅 디바이스(100)의 종래의 컴포넌트들(102 내지 116)의 각각은 데이터와 전력 신호들을 전송하기 위한 버스(118)를 통해 상호접속될 수 있다.

[0036] 위에 간략하게 논의된 바와 같이, 핸드헬드 컴퓨팅 디바이스(100)는 디스플레이(106)를 포함한다. 디스플레이(106)는 핸드헬드 디바이스에서 동작하기에 적합한 투과형 박막 액정 표시 장치 또는 기타 유형의 디스플레이를 포함할 수 있다. 또한 위에 간략하게 논의되는 바와 같이, 디스플레이(106)는 터치 스크린 기능이 구비될 수 있다. 스타일러스나 다른 포인팅 디바이스의 사용을 통해, 사용자는 디스플레이(106)의 원하는 부분을 터치하여 핸드헬드 디바이스(100)로 입력을 제공할 수 있다. 그 다음, 사용자가 터치하는 스크린의 좌표들은 프로세싱을 위해 운영 시스템(110)으로 리턴된다. 그러므로, 디스플레이는 핸드헬드로 입력과 출력 능력들을 모두 제공할 수 있음을 이해할 것이다.

[0037] 도 2a와 도 2b를 이제 참조하면, 본 발명의 다양한 실시예들에서 사용되는 디스플레이(106)에 대한 추가적 세부 사항들이 기재될 것이다. 도 2a와 도 2b에 도시된 바와 같이, 핸드헬드 내에서 사용되는 디스플레이(106)는 사용되는 디스플레이 패널의 유형에 따라 상이한 픽셀 밀도로 구성될 수 있다. 예를 들어, 도 2a에서 도시된 바와 같이, 240 수평 픽셀과 320 수직 픽셀의 픽셀 해상도를 갖는 디스플레이(106)를 위해 패널이 사용될 수 있다. 도 2a에서 도시된 디스플레이(106)의 크기(dimensions)는 수직으로 3.33 인치이고 수평으로 2.5 인치일 것이다. 이들 크기와 픽셀 해상도에서, 도 2a에 도시된 디스플레이(106)의 픽셀 밀도는 96dpi이다.

[0038] 도 2b에 도시된 디스플레이(106)는 도 2a에 도시된 디스플레이(106)보다 고 픽셀 밀도를 갖는다. 특히, 디스플레이(106)는 480 수평 픽셀과 640 수직 픽셀의 해상도를 갖고, 한편 도 2b에서 도시된 디스플레이(106)의 크기는 도 2a에 도시된 디스플레이(106)와 동일하다. 따라서, 도 2b에서 도시된 디스플레이(106)의 픽셀 밀도는 192dpi이다. 96dpi와 192dpi의 픽셀 밀도들을 갖는 디스플레이들은 단순히 설명적이고, 본 발명의 실시예들은 임의의 픽셀 밀도를 갖는 디스플레이와 함께 사용될 수 있음을 이해할 것이다.

[0039] 도 3a와 도 3b를 이제 참조하면, 추가 세부사항들은 본 발명의 실시예들에서 사용된 디스플레이들(106)에 대해 제공될 것이다. 도 3a는 96dpi 픽셀 밀도를 갖는 디스플레이의 스크린 뷰를 도시한다. 도 3a에 도시된 바와 같이, 스크린은 다수의 픽셀들(122a 내지 122c)의 수로 형성된다. 각각의 픽셀은 디스플레이 스크린에 데이터



를 쓰기 위해 개별적으로 불러질 수 있다. 예를 들어, 도 3a에 도시된 바와 같이, 사각형(120)은 적합한 픽셀들을 활성화시켜서 디스플레이 스크린 상에 디스플레이될 수 있다.

[0040] 도 3b는 192dpi의 픽셀 밀도, 또는 도 3a에서 도시된 디스플레이 스크린의 픽셀 밀도의 2배를 갖는 디스플레이 스크린(106)을 도시한다. 도 3b는 또한 도 3a에서 도시되는 동일 사각형(120)을 그리는 것을 설명한다. 사각형(120)을 그리기 위해 사용되는 픽셀들의 수가 동일하지만, 도 3b에 도시된 사각형(120)은 증가된 픽셀 밀도 때문에 도 3a에 도시된 사각형의 1/4 크기이다. 예를 들어, 이것은 저 픽셀 밀도(도 3a에 도시된 바와 같이)를 갖는 스크린과 동작하기 위해 특별히 작성된 응용 프로그램이 고 픽셀 밀도(도 3b에 도시된 바와 같이)의 디스플레이 스크린을 갖는 디바이스 상에서 실행될 때 발생할 수 있다. 본 발명의 실시예들은 도 3a에 도시된 사각형(120)과 같은 방식으로 사각형(120)이나 임의의 다른 온-스크린 객체가 도 3b에 도시된 디스플레이(106) 상에서 디스플레이되도록 할 것이다.

[0041] 도 4a와 도 4b를 이제 참조하면, 증가된 픽셀 밀도를 갖는 디스플레이 디바이스들에서 응용 프로그램 호환성을 인에이블하기 위해 사용되는 몇 개의 소프트웨어 컴포넌트들이 기재될 것이다. 특히, 도 4a는 고 픽셀 밀도 디스플레이를 갖는 디바이스에서 사용하도록 특별히 프로그램된 응용 프로그램(112a)에 의해 디스플레이 스크린 I/O를 수행하는 API들의 콜들을 프로세스하기 위해 사용되는 소프트웨어 컴포넌트들을 도시한다. 응용 프로그램(112a)은 고 픽셀 밀도 디스플레이와 함께 사용하기 위해 특별히 구성되기 때문에, API(124)로부터의 콜들은 직접 될 수 있고, 임의의 추가 프로세싱이 없이 운영 시스템(110)에 의해 프로세스될 수 있다.

[0042] 그러나, 아래 더 상세히 기재되는 바와 같이, 콜하는 응용 프로그램이 고 픽셀 밀도를 갖는 디스플레이 디바이스에서 사용하도록 특별히 구성되었는지에 대한 API 콜이 수신될 때 결정이 될 수 있다. 응용 프로그램이 고 픽셀 밀도를 갖는 디스플레이 디바이스에서 사용하기 위해 구성되면, 상술된 바와 같이 고 픽셀 밀도 디스플레이의 호환성을 인에이블하기 위해 아무런 추가 프로세싱이 필요하지 않다. 그러나, 응용 프로그램이 고 픽셀 밀도를 갖는 디스플레이와의 사용을 위해 특별히 프로그램되지 않았다면, 입력과 출력이 고 밀도 디스플레이와 호환됨을 확실하게 하기 위해 추가 프로세싱이 필요하다. 이것은 도 4b에 도시된다.

[0043] 도 4b에 도시된 바와 같이, 고 픽셀 밀도 디스플레이 디바이스에서 사용하도록 구성되지 않은 레거시 응용 프로그램(112b)은 API(124)를 콜해서 스크린 입력이나 출력 동작을 수행할 수 있다. 양방향의 변환 레이어(126)는 API 콜을 인터셉트하고, 응용 프로그램이 고 픽셀 밀도 디스플레이 디바이스에서 사용하도록 구성되었는지를 판정한다. 레거시 응용 프로그램(112b)은 고 밀도 디스플레이 디바이스에서 사용하도록 구성되지 않았으므로, 변환 레이어는 API 콜의 픽셀 파라미터들을 고 픽셀 밀도 디스플레이로 스케일한다. 예를 들어, 레거시 응용 프로그램(112b)이 96dpi 디스플레이와의 사용을 위해 구성되고, 핸드헬드(100)가 설치된 192dpi 디스플레이를 실제로 가지면, 변환 레이어(126)는 API 콜과 함께 수신된 픽셀 좌표들을 2배로 할 것이다. 그 다음, 변환 레이어(126)는 스케일된 파라미터들로 의도된 API(124)를 콜하여 고 픽셀 밀도에 요구된 동작을 수행한다. 유사하게, 예를 들어, 입력 동작이 수행될 때, 파라미터들이 API(124)로부터 리턴되면, 리턴 파라미터들은 또한 적절히 스케일될 수 있다. 도 4a와 도 4b에 도시된 API(124)는 동일하고, 아무런 변경이 그 API(124)에 될 필요가 없음을 이해해야 한다. 도 5와 그것의 관련된 논의는 API 콜들을 인터셉트하고 수신된 파라미터들을 스케일하는 변환 레이어(126)에 의해 수행되는 프로세스에 대해 추가 세부사항들을 제공한다.

[0044] 도 5를 이제 참조하면, 예시적 루틴(500)은 증가된 픽셀 밀도를 갖는 디스플레이 디바이스들에서 응용 프로그램 호환성을 인에이블하는 프로세스를 도시하여 설명될 것이다. 본 명세서에서 제공된 루틴들의 논의를 읽었을 때, 본 발명의 다수의 실시예들의 논리 동작들은 (1) 컴퓨팅 시스템 상에서 실행하는 컴퓨터 구현된 동작들 또는 프로그램 모듈들의 시퀀스로서 및/또는 (2) 상호접속된 머신 논리 회로들 또는 컴퓨팅 시스템 내의 회로 모듈들로서 구현됨을 이해할 것이다. 본 구현은 본 발명을 구현하는 컴퓨팅 시스템의 수행 요구사항들에 따른 선택 문제이다. 따라서, 도 5에 도시된 논리 동작들, 및 본 명세서에 기재된 본 발명의 실시예들을 만드는 것은 동작, 구조적 디바이스, 액트, 또는 모듈로서 다양하게 언급된다. 당업자들이라면, 이들 동작, 구조적 디바이스, 액트, 및 모듈이 첨부된 청구범위 내에서 수신된 바와 같이 본 발명의 취지 및 범위로부터 벗어나지 않고 소프트웨어, 펌웨어, 특수 목적 디지털 논리, 및 그것들의 기타 조합으로 구현될 수 있음을 인식할 것이다.

[0045] 루틴(500)은 스크린 입력이나 출력을 수행하기 위해 응용 프로그램(112)으로부터 API(124)로 전달되는 콜이 수신되는지를 변환 레이어(126)가 판정하는 동작(502)에서 시작한다. 그런 콜이 수신되지 않으면, 루틴(500)은 다른 그런 결정이 되어지는 동작(502)으로 다시 되돌아간다. 응용 프로그램(112)으로부터 API(124)로 콜이 되어지면, 루틴(500)은 동작(502)에서 동작(504)으로 계속한다.

[0046] 동작(504)에서, 변환 레이어(126)는 응용 프로그램(112)으로부터의 API 콜을 인터셉트한다. 예를 들어, 이것은

변환 레이어(126)의 주소를 주소 표의 스크린 API(124) 콜들의 각각으로 대체하여 성취될 수 있다. 이 방식으로, 스크린 API(124) 함수들로의 콜들은 대신 변환 레이어(126)로 전달된다. 그 다음, 루틴은 동작(504)에서 동작(506)으로 계속한다.

[0047] 동작(506)에서, 변환 레이어(126)는 콜하는 응용 프로그램이 증가된 픽셀 밀도 디스플레이와 함께 사용되기 위해 구성되는지를 판정한다. 변환 레이어는, 예를 들어, 응용 프로그램(112)이 컴파일되는 소프트웨어 개발 키트("SDK") 버전 번호를 결정하기 위해 응용 프로그램(112)의 특성을 조사하여, 이 프로세스를 수행할 수 있다. 이 정보는 컴파일 시에 응용 프로그램(112)에 포함될 것이다. SDK의 버전 번호에 기초하여, 변환 레이어(126)는 응용 프로그램(112)이 저 픽셀 밀도를 갖는 디스플레이 디바이스에서 사용 또는 고 픽셀 밀도를 갖는 디스플레이 디바이스에서 사용하도록 구성되었는지를 판정할 수 있다. 소정의 번호보다 크거나 같은 버전 번호들을 갖는 응용 프로그램들은 고 픽셀 밀도를 갖는 디스플레이를 지원한다고 가정된다. 소정의 번호보다 더 낮은 버전 번호들을 갖는 응용 프로그램들은 단지 저 픽셀 밀도를 갖는 디스플레이들만을 지원한다고 가정된다. 다른 방법들은 또한 응용 프로그램(112)이 고 픽셀 밀도를 갖는 디스플레이 디바이스에서 사용하도록 구성되는지를 판정하기 위해 사용될 수 있다.

[0048] 동작(506)으로부터, 루틴(500)은 응용 프로그램(112)이 고 픽셀 밀도 디스플레이 디바이스 또는 저 픽셀 밀도 디스플레이 디바이스에서 사용하도록 구성되는지에 따라 분기가 되는 결정 동작(508)으로 진행한다. 응용 프로그램(112)이 고 픽셀 밀도를 갖는 디스플레이와 함께 사용되기 위해 구성되면, 루틴(500)은 전달되는 파라미터들에 아무런 변경없이 API 콜이 의도된 API로 전달되는 동작(510)으로 진행한다. 응용 프로그램(112)은 고 픽셀 밀도 디스플레이와 사용되기 위해 구체적으로 구성되었기 때문에, API(124)로부터의 콜들은 직접 되어지고 임의의 추가 프로세싱이 없이 운영 시스템(110)에 의해 처리될 수 있다. 동작(510)에서, 루틴(500)은 추가 API 콜들이 프로세스될 수 있는 동작(502)으로 리턴한다.

[0049] 결정 동작(508)에서, 콜하는 응용 프로그램(112)이 고 픽셀 밀도를 갖는 디스플레이와 사용하기 위해 구성되지 않은 것으로 결정되면, 루틴(500)은 결정 동작(508)에서 동작(512)으로 분기한다. 동작(512)에서, API 콜은 양방향 변환 레이어(126)에 의해 인터셉트된다. 동작(514)에서, 양방향 변환 레이어(126)는 API 콜과 함께 수신된 픽셀 파라미터들을 고 픽셀 밀도 디스플레이 디바이스로 스케일한다. 그 다음, 동작(514)에서, 변환 레이어(126)는 스케일된 파라미터들로 의도된 API를 콜한다. 그 다음, API(124)는 고 픽셀 밀도 디스플레이 디바이스에 스케일된 파라미터들로 의도된 기능들을 수행한다.

[0050] 동작(514)으로부터, 예를 들어, 입력 동작이 수행될 때와 같이, 임의의 픽셀 값들이 API(124)에 의해 콜하는 응용 프로그램(112)으로 리턴해야 하는지를 변환 레이어(126)가 판정하는 동작(516)으로 루틴(500)은 진행한다. 아무런 픽셀 값이 응용 프로그램(112)으로 리턴되어야 하지 않으면, 루틴(500)은 다른 API 콜이 인터셉트될 수 있거나 프로세스될 수 있는 동작(502)으로 리턴한다. 값들이 리턴되어야 한다면, 루틴(500)은 변환 레이어가 리턴된 픽셀 값들을 저 픽셀 밀도 디스플레이의 픽셀 밀도로 스케일하는 동작(518)으로 진행한다. 그 다음, 스케일된 리턴 값들이 변환 레이어에 의해 응용 프로그램(112)으로 리턴되는 동작(520)으로 루틴(500)은 진행한다. 그 다음, 루틴(500)은 다른 API 콜이 인터셉트될 수 있고 프로세스될 수 있는 동작(502)으로 리턴한다.

[0051] 본 명세서에 기재된 본 발명의 실시예들은 핸드헬드 컴퓨팅 디바이스의 컨텍스트로 제공되지만, 본 발명의 다양한 양태들은 데스크톱, 랩톱, 및 기타 유형들의 컴퓨터 시스템들에서 사용되어 고 픽셀 밀도 디스플레이들의 백워드 호환성을 확실하게 할 수 있음을 이해해야 한다. 더욱이, 또한 본 발명의 실시예들은 저 픽셀 밀도에서 고 픽셀 밀도로 파라미터들을 스케일하는 관점으로 기재되고, 본 발명은 또한 고 픽셀 밀도들에서 저 픽셀 밀도들로 스케일하기 위해 사용될 수 있다. 이 방식으로, 고 픽셀 밀도를 갖는 디스플레이와의 사용을 위해서만 구성된 응용 프로그램들은 저 픽셀 밀도를 갖는 디스플레이 스크린 상에 수정없이 사용될 수 있다.

[0052] 상술된 바에 기초하여, 본 발명의 다양한 실시예들은 증가된 픽셀 밀도를 갖는 디스플레이 디바이스들에서 응용 프로그램 호환성을 인에이블하기 위해 방법, 시스템, 장치, 및 컴퓨터 판독가능 매체를 포함함을 이해해야 한다. 위의 명세서, 예, 및 데이터는 본 발명의 조합의 제조와 사용의 완전한 설명을 제공한다. 본 발명의 다수의 실시예들은 본 발명의 취지 및 범위로부터 벗어나지 않고 만들어질 수 있으므로, 본 발명은 이하에 첨부된 청구범위에 기재된다.

### 발명의 효과

[0053] 본 발명은 증가된 픽셀 밀도를 갖는 디스플레이 디바이스들에서 응용 프로그램의 호환성을 인에이블하기 위한 방법 및 장치를 제공한다. 본 발명에 의해 제공된 기능을 사용하여, 레거시 응용 프로그램들은 수정없이 API

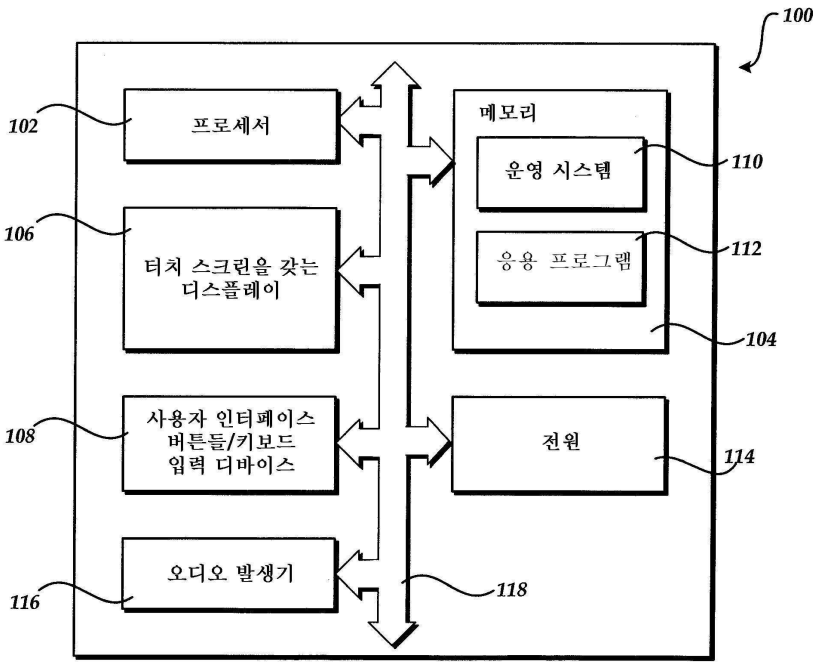
를 통해 개선된 픽셀 밀도 스크린으로 정확하게 입력하고 출력할 수 있다.

### 도면의 간단한 설명

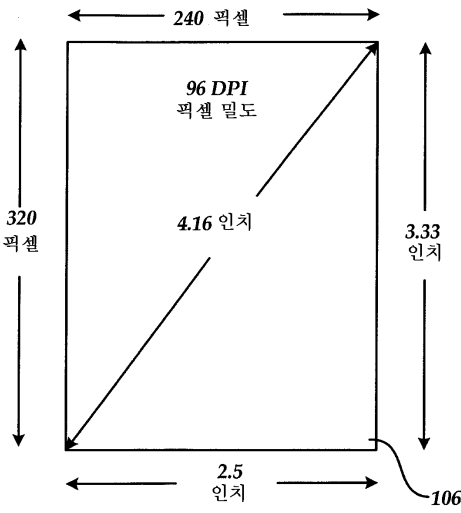
- [0001] 도 1은 본 발명의 실시예들을 위한 운영 환경을 제공하는 핸드헬드 컴퓨팅 디바이스의 아키텍처를 도시하는 컴퓨터 시스템 아키텍처도.
- [0002] 도 2a 및 도 2b는 본 발명의 다양한 실시예들에서 사용될 수 있는 2개의 상이한 디스플레이 스크린들의 해상도와 픽셀 밀도를 도시하는 블록도.
- [0003] 도 3a 및 도 3b는 종래 방법들을 사용하여 상이한 픽셀 밀도들을 갖는 2개의 상이한 디스플레이들 상에 동일 객체를 디스플레이하는 영향을 도시하는 블록도.
- [0004] 도 4a 및 도 4b는 개선된 픽셀 밀도를 갖는 디스플레이 디바이스에서 응용 프로그램의 호환성을 확실하게 하기 위해 본 발명의 다양한 실시예들에 의해 사용되는 소프트웨어 아키텍처들.
- [0005] 도 5는 본 발명의 일 구현에 따라 개선된 픽셀 밀도를 갖는 디스플레이 디바이스들에서 응용 프로그램의 호환성을 확실하게 하는 예시적 방법에 의해 수행되는 논리 동작들을 도시하는 흐름도.
- [0006] <주요 도면 부호 설명>
- [0007] 102 프로세서
- [0008] 104 메모리
- [0009] 106 터치스크린을 갖는 디스플레이
- [0010] 108 사용자 인터페이스 버튼들/키보드 입력 디바이스
- [0011] 110 운영 시스템
- [0012] 112 응용 프로그램들
- [0013] 114 전원
- [0014] 116 오디오 발생기

도면

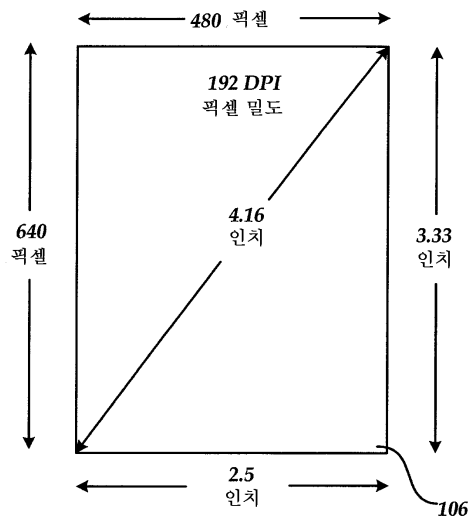
도면1



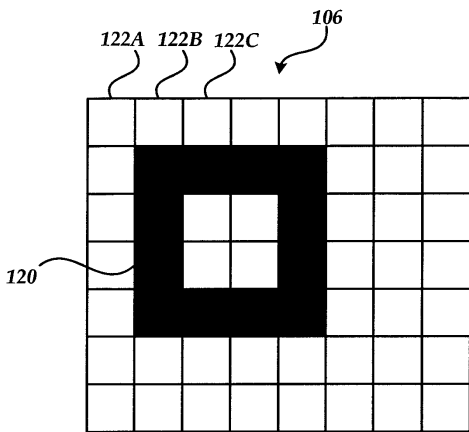
도면2a



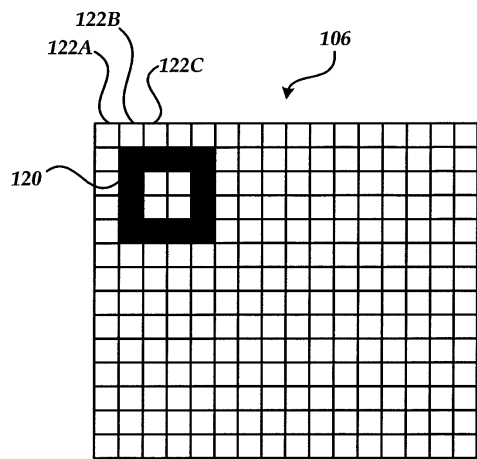
도면2b



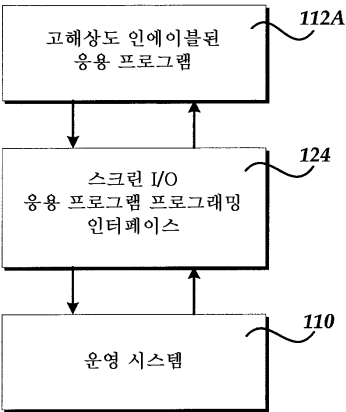
도면3a



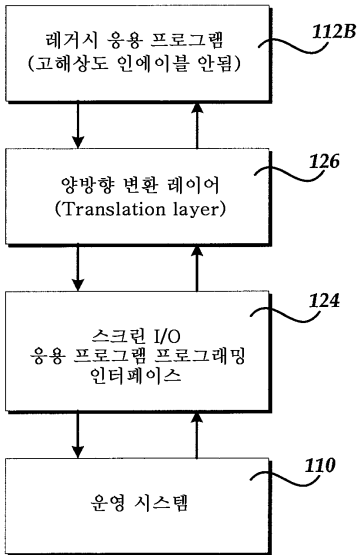
도면3b



도면4a



도면4b





도면5

