(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
22 January 2009 (22.01.2009)

PCT

(10) International Publication Number
WO 2009/009896 A1

(51) International Patent Classification:
*G06T 1/00* (2006.01) *G06T 5/00* (2006.01)
*G06T 11/00* (2006.01) *G06T 7/60* (2006.01)
*G06T 3/40* (2006.01)

(21) International Application Number:
PCT/CA2008/001311

(22) International Filing Date: 16 July 2008 (16.07.2008)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
11/779,006 17 July 2007 (17.07.2007) US

(71) Applicant *(for all designated States except US)*: **SMART TECHNOLOGIES ULC** [CA/CA]; 300, 1207-11 Avenue S.W., Calgary, Alberta T3C 0M5 (CA).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **BANERJEE, Shymmon** [CA/CA]; 175 Dalcastle Close NW, Calgary, Alberta T3A 1Z4 (CA). **KROEKER, Wallace** [CA/CA]; Box 26, Site 16, RR 8, Calgary, Alberta T2J 2T9 (CA). **BENNER, Erik** [CA/CA]; 273 Gleneagles View, Cochrane, Alberta T4C 2H6 (CA). **LEUNG, Andy** [CA/CA]; 68 Whitefield Crescent NE, Calgary, Alberta T1Y 5K1 (CA).

(74) Agents: **RUSTON, David, A.** et al.; Sim & McBurney, 330 University Avenuel|6th Floor, Toronto, Ontario M5G 1R7 (CA).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH,

*[Continued on next page]*

(54) Title: METHOD FOR MANIPULATING REGIONS OF A DIGITAL IMAGE

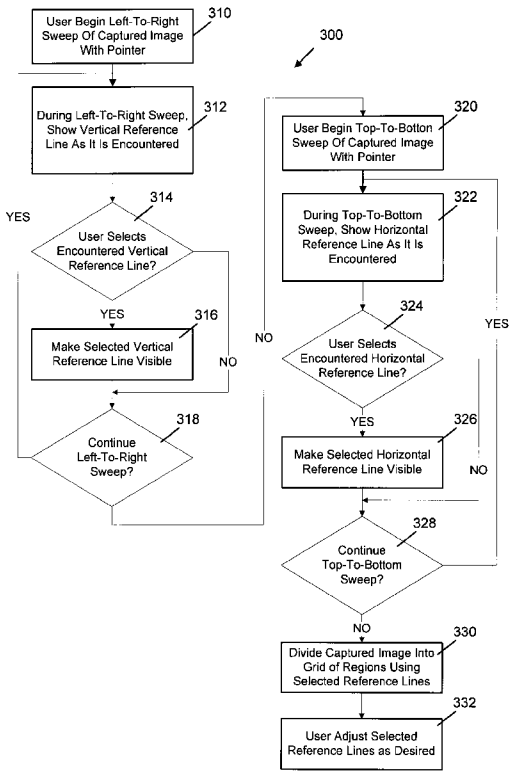



Fig. 7

(57) Abstract: A method for dividing a digital image into regions comprises identifying potential region borders based on edge content in the digital image. The digital image is divided into regions based on user-selected ones of the potential region borders. A method of processing a region of a digital image comprises receiving gesture data for characterizing the region. A processing tool associated with the gesture data is automatically launched, and the region is processed using the processing tool.

WO 2009/009896 A1

GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MT, NL, NO, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

**Published:**
— *with international search report*

# METHOD FOR MANIPULATING REGIONS OF A DIGITAL IMAGE

## Field of the Invention

**[0001]** The present invention relates generally to image processing, and more particularly to a method and apparatus for manipulating regions of a digital image.

## Background of The Invention

**[0002]** In a newsroom or other broadcast environment, it is often desirable to transmit digital images taken from a variety of sources as part of a newscast, a sportscast, or for advertising. It is increasingly common for such transmitted digital images to include World Wide Web (WWW) pages or portions thereof that have been screen captured from the Internet using a personal computer, computer workstation or the like.

**[0003]** As is well known, a Web page often contains several visual and functional elements, including columns of text, advertisements, images, hyperlinks, live stock quote tickers, embedded programs, and the like. In the case where it is desired to broadcast a screen-captured image of a Web page for display on television screens, the image is typically manually edited prior to transmission. This is done to remove regions of the image that are not relevant to the subject-matter of the story, and/or to scale regions of the image such that they are more readable on the television screen once transmitted. For example, a newscaster may wish to draw the attention of a television audience to a particular quote on a particular Web page, but remove advertisements and other extraneous items.

**[0004]** It can be time-consuming to manually identify, isolate and manipulate regions in an image of a Web page. For this purpose, production staff typically employ an image cropping tool that is packaged with a set of editing tools as part of an image editing application (such as Microsoft™ Paint, for example). During image manipulation, image regions are manually copied from the original Web page image and pasted into a separate image for combining with other image elements. In order to ensure that the final image is attractive and useful, it is advantageous to identify and isolate image regions with pixel accuracy. To do so typically requires that production staff sequentially select regions and employ zoom-in, zoom-out, cropping and perhaps other digital image processing tools. Careful control of the pointer (i.e., mouse, stylus etc.) to identify the bounds of each selected image region in the digital image is required.

**[0005]** Techniques to crop images have been considered. For example, U.S. Patent No. 6,337,925 to Cohen et al. discloses a method for determining borders of objects in a digital image in order to assist with masking of an object bound by the border. A user is required to choose an area of interest of the digital image that includes a portion of a border to be identified. The border portion is then modeled by estimating a position, direction and width of the edge zone of the border. Based on the modeled border portion, the border of an entire object in the scene is identified.

**[0006]** U.S. Patent No. 6,593,944 to Nicolas et al. discloses a method and electronic system for modifying a Web page such that its content can be read by a user on an electronic device having a small screen. The method presupposes that the Web page is divided into frames, and arranges the Web page such that the frames can be displayed individually on the small screen. Each frame is scaled to the full size of the display.

**[0007]** Although manipulating digital images prior to transmission is often required as described above, in some instances it is also desired to manipulate broadcasted images. For example, during a live television broadcast, it is often desirable to have images available to the commentator for real-time display and further manipulation. In order to provide emphasis during a newscast, a commentator may wish to spotlight an image region that includes a popular quote. As another example, during a sportscast a commentator may wish to encircle an image region showing a goal being scored in a hockey game, then zoom in on the region to show the puck crossing the goal line.

**[0008]** In order to achieve real-time digital image processing, the commentator may employ a computer workstation directly, or some other user interface such as a touch system. Touch systems are well known in the art and typically include a touch screen having a touch surface on which contacts are made using a pointer in order to generate user input. The user input is then automatically conveyed to a computer executing one or more application programs. The computer uses the user input to update the image being presented on the touch screen, or to perform other actions via the application programs as though they were being operated using mouse and keyboard, for example. Many types of touch systems exist that utilize disparate technologies to identify contacts with the touch surface. These technologies include

for example analog resistive, surface acoustic wave, capacitive, infrared, electromagnetic, laser-based and machine-vision detection arrangements to identify contacts with the touch surface.

**[0009]** In order to select a region and a processing tool (such as zoom-in, zoom-out, spotlight, highlight, or crop, as a few examples), the commentator typically pre-selects the processing tool application from a menu, picks up a pointer, and draws a border around the region of interest in the image displayed on the touch screen. The processing tool application running on the computer, which receives the user input, performs the desired processing action on the selected image region, and updates the image presented on the touch screen. Alternatively, the commentator may first select the region in the displayed image with the pointer and then select the desired processing tool application from a menu. Once selected, the processing tool application performs its processing action on the selected region, and the image presented on the touch screen is updated.

**[0010]** The manual selection of processing tools and identification of image regions can be time-consuming, particularly in the context of a live broadcast. This concern is addressed somewhat by application programs that permit users to first identify an image region and then select particular image processing tools using specific pointer gestures (such as an up-right motion indicating zoom, and a right-left motion indicating highlight). Other applications require receipt of a particular sequence of pointer gestures to perform various functions. For example, the video game Black & White, (developed by Lionhead Studios and published by Electronic Arts and Feral Interactive), allows players to cast spells and perform miracles by forming shapes using a sequence of mouse movements. The Web browser Opera enables a user to perform browsing gestures, as opposed to image processing operations, using small, quick mouse movements.

**[0011]** It is known to assign a different processing tool application to each of a set of pointers. With such an approach, the processing tool application associated with the selected pointer is automatically invoked when input is generated using that pointer. For example, a user may use a zoom pointer to identify an image region upon which zoom processing is to be conducted.

**[0012]** Although techniques to manipulate images prior to and after transmission

are known as described above, improvements are desired. It is therefore an object of the present invention to provide a novel method and apparatus for dividing a digital image into regions for individual manipulation and for performing an image processing operation on a region in a digital image.

**Summary of the Invention**

**[0013]** In accordance with one aspect, there is provided a method of dividing a digital image into regions, comprising:

identifying potential region borders based on edge content in the digital image; and

dividing the digital image into regions based on user-selected ones of the identified potential region borders.

**[0014]** In one embodiment, the potential region borders are identified at locations along rows and columns in the digital image that have a significant amount of edge content. The identifying comprises calculating pixel intensity profiles for each row and column of pixels in the digital image. Adjacent ones of the pixel intensity profiles are compared to determine where differences between adjacent pixel intensity profiles exceed a threshold level, thereby establishing row and column positions of the potential region borders.

**[0015]** In accordance with another aspect, there is provided a method of processing a region of a digital image, comprising:

receiving gesture data for characterizing the region;

automatically launching a processing tool associated with the gesture data; and

processing the region using the processing tool.

**[0016]** In accordance with another aspect, there is provided a computer readable medium embodying a computer program for dividing a digital image into regions, the computer program comprising:

computer program code identifying potential region borders based on edge content in the digital image; and

computer program code dividing the digital image into regions based on user-selected ones of the identified potential region borders.

**[0017]** In accordance with yet another aspect, there is provided a computer

readable medium embodying a computer program for processing a region of a digital image, the computer program comprising:

computer program code receiving gesture data for characterizing the region;

computer program code automatically launching a processing tool associated with the gesture data; and

computer program code processing the region using the processing tool.

**[0018]** In accordance with another aspect, there is provided an interactive display system comprising:

a touch screen that defines a touch surface on which a computer-generated image is presented;

a sensor assembly detecting pointer contacts on the touch surface and in response, generating gesture data for characterizing a region of the computer-generated image; and

processing structure receiving the gesture data and automatically launching a processing tool associated with the gesture data for processing the region.

**[0019]** The above method for dividing a digital image into regions is beneficial as it provides for the accurate identification and manipulation of image regions during editing of the digital image. A user is able to select from pre-defined region borders to divide the image into regions and is thereby not necessarily required to laboriously define the image regions manually. As such, time spent by a user defining image regions and editing the image using the image regions, particularly for digital images having certain types of content (i.e. Web pages), is significantly reduced.

**[0020]** The method for processing a region of a digital image is beneficial as it provides for the accurate identification and processing of image regions, particularly during live broadcasting of the images. A user is able to identify both a region of interest, and the processing tool for processing the region, with a single gesture. As such, time spent by a user identifying the image region and selecting a processing tool is significantly reduced.

## Brief Description of the Drawings

**[0021]** Embodiments will now be described more fully with reference to the accompanying drawings, in which:

**[0022]** Figure 1 is a flowchart showing steps for dividing a digital image into regions;

**[0023]** Figure 2 is a flowchart showing the steps performed during identification of potential region borders in the digital image;

**[0024]** Figure 3 is a captured image of a Web page;

**[0025]** Figure 4 is a grayscale image of the Web page of Figure 3, showing computed column and row intensity profiles;

**[0026]** Figure 5 is the grayscale Web page image of Figure 4, showing thresholded column and row intensity profiles to identify locations of reference lines;

**[0027]** Figure 6 shows the identified reference lines overlaid on the captured Web page image;

**[0028]** Figure 7 is a flowchart showing steps for selecting and manipulating reference lines;

**[0029]** Figure 8 shows the captured Web page image of Figure 3 from a user's point of view, in which a vertical sweep tool for selecting from the identified reference lines is invoked;

**[0030]** Figure 9 shows the captured Web page image of Figure 8 after vertical and horizontal sweeps for selecting reference lines has been completed;

**[0031]** Figure 10 shows the captured Web page image of Figure 8 after a subset of image regions have been discarded;

**[0032]** Figure 11 shows degrees of freedom for manipulating the sizes of undiscarded image regions of the Web page image of Figure 10;

**[0033]** Figure 12 shows the undiscarded image regions of the Web page image of Figure 10 having been manipulated to form a complete, simplified Web page image;

**[0034]** Figure 13 is a front plan view of a touch system;

**[0035]** Figure 14 is a flowchart showing steps for processing a region of a digital image using the touch system;

**[0036]** Figure 15a shows a region of a digital image delineated by a generally rectangular boundary drawn on a touch surface of the touch system;

**[0037]** Figure 15b shows the delineated region of the digital image of Figure 15a enlarged by application of a zoom tool associated with the drawn rectangular boundary;

**[0038]** Figure 16a shows a region of another digital image delineated by a generally circular boundary drawn on the touch surface of the touch system; and

**[0039]** Figure 16b shows the delineated region of the digital image of Figure 16a having been spotlighted by application of a spotlighting tool associated with the drawn circular boundary.

**Detailed Description of the Embodiments**

**[0040]** In the following description, methods, apparatuses and computer readable media embodying computer programs for dividing a digital image into regions and processing a region of a digital image are disclosed. The methods and apparatuses may be embodied in a software application comprising computer executable instructions executed by a processing unit including but not limited to a personal computer, interactive display or touch system, a digital image or video capture device such as for example a digital camera, camcorder or electronic device with video capabilities, or other computing system environment. The software application may run as a stand-alone digital image tool, an embedded function or may be incorporated into other available digital image applications to provide enhanced functionality to those digital image applications. The software application may comprise program modules including routines, programs, object components, data structures etc. and may be embodied as computer readable program code stored on a computer readable medium. The computer readable medium is any data storage device that can store data, which can thereafter be read by a computer system. Examples of computer readable media include for example read-only memory, random-access memory, CD-ROMs, magnetic tape and optical data storage devices. The computer readable program code can also be distributed over a network including coupled computer systems so that the computer readable program code is stored and executed in a distributed fashion. Embodiments will now be described with reference to Figures 1 to 15.

**[0041]** Turning now to Figure 1, a method of dividing a digital image into regions is shown. During the method, the digital image is captured (step 100) as a computer screenshot or using an image capture device (such as a digital camera, for example). Potential or candidate region borders within the digital image are then identified as reference lines (step 200), and user selected reference lines are determined (step 300).

Based on the selected reference lines, the captured image is divided into regions (step 400) for further manipulation by the user.

**[0042]** The steps for identifying potential region borders (step 200) are shown in Figure 2. First, a grayscale image is created by extracting the pixel luminance channel from the captured image (step 210). Intensity profiles are then computed for each row and column of the grayscale image (step 212). During computation of the intensity profiles, the luminance values of the pixels in each row of the grayscale image are averaged thereby yielding, for each row, an intensity profile that is the average luminance of all pixels in the row. The same is performed for the columns of the grayscale image.

**[0043]** After computation of the intensity profiles, the intensity profiles for the rows are thresholded to identify the locations of significant horizontal edges in the grayscale image, as set forth in the following pseudo code:

```
If | IntensityProfile(row (i-1)) – Intensity Profile(row(i)) | >= Threshold
    IntensityProfile(row(i)) = 1
Else
    IntensityProfile(row(i)) = 0
End If
```

**[0044]** A fixed threshold value of 0.125 intensity units has been found to be suitable. An adaptive threshold value or one obtained based on a measurement of intensity volatility in the grayscale image (i.e. standard deviation of intensity, for example) may of course be employed.

**[0045]** The intensity profiles for the columns are then thresholded in a manner similar to that described above for the rows.

**[0046]** With the intensity profiles thresholded, the positions of intensity profiles having a designation of "1" correspond to positions of region borders or reference lines in the captured image. Accordingly, horizontal reference lines are identified in the captured image at rows having an intensity profile = 1 (step 214), and vertical reference lines are identified in the captured image at columns having an intensity profile=1 (step 216). Each horizontal reference line spans the entire width of the captured image, and each vertical reference line spans the entire height of the captured image.

**[0047]** Figures 3 to 6 show processing of a Web page image according to the steps described above. More particularly, Figure 3 shows a captured Web page image as it would appear to a user. Figure 4 shows the corresponding grayscale image and the computed column and row intensity profiles. It will be understood that, while intensity profiles are visible in Figure 4, this is for ease of understanding. It is not necessary for the intensity profiles to be displayed. Figure 5 illustrates the column and row intensity profiles of the grayscale image having been thresholded to identify the vertical and horizontal reference lines. Figure 6 shows the reference lines overlaid on the captured Web page image. It will be understood that, while the reference lines are visible in Figure 6, this is for ease of understanding. As will be described in further detail below, according to this embodiment it is not necessary at this point for users to see them displayed onscreen simultaneously.

**[0048]** With the horizontal and vertical reference lines identified, the user is able to select reference lines in order to divide the captured image into regions for individual manipulation (step 300). The steps performed during reference line selection are shown in Figure 7, as described below.

**[0049]** In order to view and select vertical reference lines, a vertical sweep tool is invoked. Once invoked, the user begins a left-to-right sweep of the captured image with the pointer (step 310). During the left-to-right sweep, the vertical reference lines are made visible to the user as they are "encountered" by the pointer (step 312). More particularly, a vertical reference line appears on the display when the x-position of the pointer is determined to match or at least be within a predetermined distance of the x-coordinate of that vertical reference line. The vertical reference line remains visible as the user continues to sweep the pointer past the x-coordinate of the vertical reference line and disappears when the actual x-coordinate of the pointer exceeds the vertical reference line x-coordinate by a threshold amount (i.e. 10 pixels, for example). This facility eases selection by the user of the vertical reference line. Furthermore, where the display includes a pointer icon (i.e., an arrow), the arrow "sticks" to the vertical reference line. This "sticking" effect also assists the user with identifying and selecting the vertical reference line. If a visible reference line is to be selected, the user simply needs to temporarily halt the sweep and click the mouse button to select the vertical reference line that has been made temporarily visible.

**[0050]** After the user selects the vertical reference line (step 314), the vertical reference line is toggled "ON" and remains permanently visible, despite the pointer's position (step 316). If the user desires to continue the left-to-right sweep to select additional vertical reference lines (step 318), then the method continues from step 312. While the vertical sweep tool is selected, the user is able to go back and re-sweep to select further vertical reference lines that had previously been missed.

**[0051]** In order to view and select horizontal reference lines, the user invokes a horizontal sweep tool and begins a top-to-bottom sweep of the captured image with the pointer (step 320) to encounter and select from horizontal reference lines. The top-to-bottom sweep and selection of horizontal reference lines is conducted in a similar manner to that described above with respect to vertical reference lines. Once the user has completed the top-to-bottom sweep, each selected horizontal reference line is toggled "ON" and remains visible, despite the pointer's position. While the horizontal sweep tool is selected, the user is able to go back and re-sweep to select further horizontal references lines that had previously been missed.

**[0052]** Should the user desire, a selected reference line can be deselected and thereby toggled "OFF" to render it invisible again.

**[0053]** Once the horizontal and vertical reference lines have been selected, the captured image is divided into a grid of regions, delineated by the selected reference lines (step 330). The user can then adjust the selected reference lines to fine-tune their positions as desired (step 332). For example, one of the horizontal reference lines in Figure 6 runs through the Headline "Microsoft Releases Windows CE 6 beta". If this horizontal reference line, once selected, were not adjusted, the Headline would be undesirably cropped. Thus, the user adjusts the selected reference line as if resizing a tiled window. Figures 9 to 12 show the selected reference line having been adjusted to ensure that the Headline is not undesirably cropped. The reference lines are divided automatically into segments based on intersection points of horizontal and vertical reference lines, and the user can selectively adjust individual segments of the reference lines thereby to adjust the borders of individual image regions.

**[0054]** Once the captured image has been divided into the desired image regions by the adjusted reference lines, the user can select individual image regions for further manipulation. For example, the user may wish to discard some image regions and

reposition and/or resize other image regions. An image region may be discarded by selecting the image region using the pointer and pressing the "delete" key on a keyboard, for example. Once selected image regions have been discarded, size adjustment of the remaining image regions can be performed by selecting with a pointer a horizontal or vertical region border and dragging it using the pointer through empty space (i.e. where discarded image regions were once positioned) to the closest border of the next undiscarded image region. If the selected border is dragged so as to overlap the next undiscarded image region, it is "snapped" into alignment with the closest border of that undiscarded image region. As such, the image region is made to neatly align with the next undiscarded image region. For vertical image region borders, the "snapping" functionality is facilitated by detecting when the x-coordinate of the selected image region border while being dragged becomes equal to or greater than the x-coordinate of the facing image region border of the next undiscarded image region. Upon release the x-coordinate of the selected image region border is re-set to be equal to the x-coordinate of the facing image region border. The same is done with horizontal borders by detecting and re-setting y-coordinates in a similar manner.

**[0055]** Upon "snapping" of the selected image region border into alignment with the closest border, the now-adjacent image regions may, as desired, be merged as a single region. Alternatively, the aligned borders themselves may be merged as a single border such that adjustment of the single border enlarges one of the two adjacent image regions and diminishes the other.

**[0056]** Figures 8 to 12 show processing of the captured Web page of Figure 3 according to the above steps when a user selects from the set of reference lines to identify image regions. More particularly, Figure 8 shows the Web page image from a user's point of view, in which the vertical sweep tool for selecting reference lines has been invoked. Figure 9 shows the Web page image after vertical and horizontal sweeps for selecting reference lines is complete. Figure 10 shows the Web page image after a user has discarded a subset of the image regions. Figure 11 shows undiscarded image regions of the Web page image of Figure 10 being manipulated thereby to adjust their respective sizes.

**[0057]** Figure 12 shows the undiscarded image regions of the Web page image of Figure 10 having been manipulated as described above to form a complete, simplified

Web page image suitable for broadcasting.

**[0058]** Although manipulating digital images prior to transmission is often required as described above, in some instances it is also desired to spotlight, zoom-in on, or otherwise manipulate regions of digital images during, for example, a live television broadcast. Manipulation of a digital image during a live television broadcast may be performed using an interactive display, or touch system 50 such as that shown in Figure 13. The touch system is similar to that described in United States Patent Application Serial No. 11/331,448, assigned to SMART Technologies Inc., assignee of the subject application, the content of which is incorporated herein by reference in its entirety. As can be seen, touch system 50 includes a touch screen 52 having a touch surface 54 defining a region of interest on which pointer contacts are to be made using pointer 70. In this embodiment, pointer 70 has a body 72 and a tip 74, and the touch screen 52 is the generally planar surface of a flat panel display device such as for example an LCD, plasma, HDTV or other television display device. A sensor assembly 56 extends along one side of the touch screen 52. The sensor assembly 56 includes a valence 58 secured to one side edge of the touch screen 52. Digital cameras 60 are positioned adjacent opposite ends of the valence 58. The fields of view of the digital cameras 60 overlap over the entire active area of the touch surface 54 so that pointer contacts made on the touch surface 54 can be visually detected.

**[0059]** An infrared (IR) receiver 62 is positioned adjacent to and communicates with an associated digital camera 60. Each IR receiver 62 is similar to those found on consumer electronics and comprises a lensed IR detector coupled to a gain controlled amplifier. The digital cameras 60 are coupled to a computer 64 or other suitable processing device via a USB hub 65 and high speed data bus 66 such as for example USB-2. Computer 64 includes memory and a processor that executes one or more application programs and provides display output that is made visible on the touch screen 52. The touch screen 52, computer 64 and display device form a closed-loop so that pointer contacts with the touch screen 52 can be recorded as writing or drawing, and/or as gestures used to control execution of application programs executed by the computer 64.

**[0060]** Figure 14 is a flowchart showing steps for selecting and processing an

identified region of a digital image during a live television broadcast with the touch system of Figure 13. A software-accessible processing tool table or tables associating shapes of gestures with processing tools is stored in memory of computer 64 (step 600). An example of the conceptual contents of the processing tool table is shown in Table 1 below.

| Gesture | Processing Tool |
|---|---|
| "O" | Spotlight (Spotlight region characterized by size and position of "O" shaped gesture) |
| "□" | Zoom (Zoom in/out on region characterized by size and position of "□" shaped gesture) |

Table 1

**[0061]** In Table 1, an elliptical-shaped gesture is associated with a "Spotlight" processing tool that brightens an identified region relative to its surroundings. Similarly, a rectangular-shaped gesture is associated with a "Zoom" processing tool.

**[0062]** When the user gestures on the touch screen 52 to identify a region of the displayed digital image, gesture data is received by computer 64 (step 700), and gesture ink is displayed on the display device in a known manner based on the received gesture data. The gesture data is a set of sampled coordinates based on pointer contacts that in combination form a closed figure such as an ellipse, a rectangle, a triangle etc. Individual sampled coordinates are considered part of a set if together received while the pointer remains in contact with the touch surface 54. Other similar criteria may be used. In this embodiment, the gesture data directly characterizes the identified region because it defines the shape, size and position of the region.

**[0063]** Once the gesture data is received, the shape of the gesture is determined (step 800) by the computer 64 using a shape recognition algorithm. The gesture

shape, as described above, is considered a closed figure (i.e. one of a rectangle, ellipse, triangle etc.). Error handling algorithms may be employed by computer 64 to handle cases in which the gesture data cannot reasonably be considered to define a closed figure (i.e. a straight or only slightly curved line). Other pre-processing algorithms may be employed by computer 64 to handle cases in which the gesture data, while not inherently defining a closed figure, may reasonably be considered to represent a closed figure. For example, a "C" shape or swirl may each be considered as imprecisely-drawn ellipses. It is advantageous, from a user's perspective, for the shape recognition algorithm to be robust so as to handle such partially-closed figures. In order to achieve this, the pre-processing algorithms process the set of sampled points and perhaps add additional points to the set to result in a closed figure that completely encloses the region being delineated.

**[0064]** Shape recognition algorithms are known in the art, and include those that compare the characteristics of a set of sampled coordinates to predefined characteristics respectively defining shapes or shape templates. For example, if a set of coordinates is a 93% match with a given shape template but only a 22% match with another shape template, then the shape template with the 93% match is chosen as the matching gesture template, thereby to determine the gesture shape.

**[0065]** Once the gesture shape is determined, the original gesture ink on the display device is removed and replaced with gesture ink corresponding to the determined gesture shape scaled and positioned so as to correspond to the size and position of the user's original gesture. The size and position of the gesture are determined by processing the set of sampled points using geometrical analysis to determine a center of the gesture and its boundaries.

**[0066]** The gesture shape is then used by the computer 64 as a key into the index of the processing tool table to identify the associated processing tool. In Table 1, for example, should the gesture shape be determined to be an ellipse "O", the Spotlight processing tool is identified.

**[0067]** The identified processing tool is then automatically launched and both the digital image and the gesture data are provided as input arguments to the processing tool for processing the region (step 1000). Processing tool input arguments may include the gesture data itself, or the determined gesture shape with corresponding

size and position information. Depending upon the processing tool with which the gesture shape is associated in the processing tool table, processing may occur automatically, or alternatively an option given to the user to specify further the nature of the processing. For example, if the processing tool is a zoom tool, the user is given the option of zooming in, zooming out, or doing nothing. Furthermore, the user is able to translate the zoom tool so as to process a region having the same shape and size as was drawn but at a different position. Once used, the processing tool is closed or otherwise de-activated by clicking the mouse or contacting the touch surface 54 outside of the region delineated by the gesture ink.

**[0068]** A region 82 of a digital image 80 delineated by a generally rectangular boundary drawn using the touch system 50 is shown in Figure 15a. The identified region 82 of digital image 80 enlarged by application of a zoom processing tool associated with the gestured rectangular boundary is shown in Figure 15b. The original gesture ink of Figure 15a is automatically removed and, as shown in Figure 15b, is replaced with gesture ink identifying the bounds of the rectangular region 82 that was enlarged by the zoom processing tool.

**[0069]** A region 92 of different digital image 90 delineated by a circular/elliptical boundary drawn by a user using the touch system 50 is shown in Figure 16a. The identified region 92 of digital image 90 spotlighted by application of a spotlighting tool associated with the gestured circular/elliptical boundary is shown in Figure 16b. The original gesture ink of Figure 16a is automatically removed and, as shown in Figure 16b, is replaced with gesture ink identifying the bounds of the circular/elliptical region 92 that was spotlighted by the spotlighting processing tool.

**[0070]** While a novel method of processing an identified region of a digital image has been described above, it will be understood that many alternatives are available. For example, other figure shapes, such as triangles or trapezoids may be included in the processing tool table in association with respective processing tools.

**[0071]** It will be understood that the gesture shapes stored in the processing tool table previously described are those shapes that serve to directly characterize a region (i.e., clearly identify its size, shape and position), or can be reasonably assumed by implementing software to clearly characterize the region. For example, an ellipse inherently encloses a region thereby fully characterizing the region, whereas a "C"

shape does not inherently characterize a region but may be considered by the implementing software to be a similarly-sized ellipse that has been incompletely drawn. The implementing software would then consider the "C" shape to characterize a region that would have been enclosed by a similarly-sized ellipse, and the "C" and ellipse would, for the purpose of processing tool selection, be the same gesture and therefore result in selection of the same processing tool.

**[0072]** According to another embodiment, the processing tool table is configured such that both a "C" and an ellipse "O", while characterizing the same region (in terms of shape, size and position), respectively cause the selection of a different processing tool for processing the region. For example, drawing a "C" shape causes an elliptical region to be spotlighted, whereas drawing an actual ellipse "O" causes the elliptical region to be cropped. Other gestures that do not directly characterize regions, but can be respectively associated with a closed figure and specify its size and position, are supported. For example, a "+" sign gesture in this embodiment characterizes a similarly-sized rectangular region as would an actual rectangular gesture surrounding the region, but causes the selection of a different processing tool than the drawing of a rectangle. The "+" sign gesture could alternatively characterize a similarly-sized elliptical region. Therefore, it can be seen that the shape, size and position of a particular region may be characterized using several different gesture shapes, whereby the actual gesture shape is the key into the index of the processing tool table. An example of the conceptual contents of such an alternative processing table is shown in Table 2 below.

| Gesture | Region Shape, Size and Position | Processing Action |
|---|---|---|
| "+" | "O" shaped region defined by size and position of "+" gesture | Crop region |
| "O" | "O" shaped region defined by size and position of "O" gesture | Zoom in/out of region |
| "□" | "□" shaped region defined by size and position of "□" gesture | Zoom in/out of region |
| "┌" | "□" shaped region defined by size and position of "┌" gesture | Highlight region |

Table 2

**[0073]** According to the contents of Table 2, if a "+" is drawn by a user, it signifies that an "O", or elliptical-shaped region corresponding in size and position to the drawn "+" is to be processed using the Crop processing tool. However, if an "O" is drawn by the user, it signifies that the same elliptical-shaped region corresponding in size and position to the drawn "O" is to be processed, but this time using the Zoom in/out processing tool.

**[0074]** According to another embodiment, gesture shapes drawn using dashed lines invoke a different processing tool than gesture shapes of the same size and position drawn using a solid line. An example of the conceptual contents of such an alternative processing table is shown in Table 3 below.

| Gesture | Region Shape, Size and Position | Processing Action |
|---|---|---|
| Dashed "O" | "O" shaped region defined by size and position of dashed "O" gesture | Crop region |
| Solid "O" | "O" shaped region defined by size and position of solid "O" gesture | Zoom in/out of region |
| Solid "+" | "O" shaped region defined by size and position of solid "+" gesture | Highlight region |
| Dashed "□" | "□" shaped region defined by size and position of dashed "□" gesture | Crop region |
| Solid "□" | "□" shaped region defined by size and position of solid "□" gesture | Zoom in/out of region |
| Solid "┌" | "□" shaped region defined by size and position of solid "┌" gesture | Highlight Region |

Table 3

**[0075]** Alternatively, the dashed lines differentiate between region shapes, size and position but invoke the same tool. For example, a dashed "+" is used to identify a circle and a cropping processing action, whereas a solid "+" is used to identify a square and the cropping processing action. Other alternatives will be apparent to those of skill in the art. For example, in some embodiments the determined gesture shape can be dependent on the orientation of the sampled points.

**[0076]** According to another embodiment, a special gesture pen may be used for drawing gesture shapes on the touch screen and launching corresponding processing tools/applications, whereas other pens are used simply for drawing.

**[0077]** It can be seen that the above described method for processing a region of a digital image advantageously allows a user to both select a region and a tool for operating on the region with a single gesture.

**[0078]** Although particular embodiments have been described above, those of skill in the art will appreciate that variations and modifications may be made without departing from the spirit and scope thereof as defined by the appended claims.

**What is claimed is**:

1. A method of dividing a digital image into regions, comprising:

identifying potential region borders based on edge content in the digital image; and

dividing the digital image into regions based on user-selected ones of the identified potential region borders.

2. The method of claim 1, wherein the potential region borders are identified at locations along rows and columns in the digital image that have a significant amount of edge content.

3. The method of claim 2, wherein said identifying comprises:

calculating pixel intensity profiles for each row and column of pixels in the digital image; and

comparing adjacent ones of the pixel intensity profiles to determine where differences between adjacent pixel intensity profiles exceed a threshold level, thereby establishing row and column positions of the potential region borders.

4. The method of claim 3, wherein the threshold level is fixed.

5. The method of claim 4, wherein the threshold level is 0.125 intensity units.

6. The method of claim 3, wherein the threshold level is calculated based on a measurement of intensity volatility in the digital image.

7. The method of claim 6, wherein the intensity volatility is the standard deviation of pixel intensity values in the digital image.

8. The method of claim 1, further comprising:

receiving a selection of a subset of the regions; and

removing the content of unselected regions from the digital image.

9. The method of claim 8, further comprising:
scaling selected regions in response to user instructions.

10. The method of claim 9, wherein said scaling comprises:
receiving a user selection of a region border; and
in response to user instructions moving the region border thereby to scale the region.

11. The method of claim 1, comprising:
modifying a position of at least one potential region border in response to user instructions.

12. The method of claim 1, further comprising:
merging regions in response to user instructions.

13. The method of claim 12, wherein said merging comprises:
receiving a selection of at least two adjacent regions to be merged;
removing a portion of any potential region borders that divide the at least two adjacent regions; and
merging the at least two adjacent regions to form a single continuous region.

14. A method of processing a region of a digital image, comprising:
receiving gesture data for characterizing the region;
automatically launching a processing tool associated with the gesture data; and
processing the region using the processing tool.

15. The method of claim 14, wherein the gesture data defines a gesture shape that is a closed figure thereby to characterize the shape, size and location of the region.

16. The method of claim 14, wherein the gesture data defines a gesture shape, the gesture shape associated with a shape of, and defining the size and position of, a closed figure thereby to characterize the region.

17. The method of claim 14, wherein the gesture data defines a gesture shape, the gesture shape being associated with the processing tool.

18. The method of claim 17, wherein the gesture shape is a closed figure.

19. The method of claim 14, wherein the gesture data includes coordinates along the boundaries of the region.

20. The method of claim 14, wherein said automatically launching a processing tool comprises:

identifying a gesture shape from the gesture data;

searching a processing tool table to determine a processing tool associated in the processing tool table with the identified gesture shape;

automatically launching the processing tool; and

providing the gesture data to the processing tool.

21. The method of claim 20, wherein identifying the gesture shape comprises:

comparing the gesture data to a plurality of shape templates each representing a unique shape; and

determining which of the shape templates is the most similar to the region thereby to identify the gesture shape.

22. The method of claim 14, wherein the processing tool is selected from the group consisting of a zoom tool, a cropping tool, a highlighting tool, and a spotlighting tool.

23. The method of claim 20, wherein the gesture shape is a rectangle and the processing tool associated with the rectangle in the processing tool table is a zoom tool.

24. The method of claim 20, wherein the gesture shape is an ellipse and the

processing tool associated with the ellipse in the processing tool table is a highlight/spotlight tool.

25. The method of claim 14, wherein processing of the region by the processing tool is controllable by a user.

26. The method of claim 25, wherein the processing tool is a zoom tool and the magnification level of the zoom tool is further controllable by the user.

27. A computer readable medium embodying a computer program for dividing a digital image into regions, the computer program comprising:

computer program code identifying potential region borders based on edge content in the digital image; and

computer program code dividing the digital image into regions based on user-selected ones of the identified potential region borders.

28. A computer readable medium embodying a computer program for processing a region of a digital image, the computer program comprising:

computer program code receiving gesture data for characterizing the region;

computer program code automatically launching a processing tool associated with the gesture data; and

computer program code processing the region using the processing tool.

29. An interactive display system comprising:

a touch screen that defines a touch surface on which a computer-generated image is presented;

a sensor assembly detecting pointer contacts on the touch surface and in response, generating gesture data for characterizing a region of the computer-generated image; and

processing structure receiving the gesture data and automatically launching a processing tool associated with the gesture data for processing the region.

30. The system of claim 29, further comprising:

memory storing a processing tool table associating at least one processing tool with a respective gesture shape;

wherein the processing structure identifies a gesture shape from the gesture data, and selects an associated processing tool from the processing tool table for processing the region.

31. The system of claim 29, wherein the processing tool is selected from the group consisting of a zoom tool, a cropping tool, a highlighting tool, and a spotlighting tool.

32. The system of claim 30, wherein the gesture shape is a rectangle and the processing tool associated with the rectangle in the processing tool table is a zoom tool.

33. The system of claim 30, wherein the gesture shape is an ellipse and the processing tool associated with the ellipse in the processing tool table is a highlight/spotlight tool.

34. The system of claim 30, wherein processing of the region by the processing tool is controllable by a user.

35. The system of claim 30, wherein the processing tool is a zoom tool and the magnification level of the zoom tool is further controllable by the user.

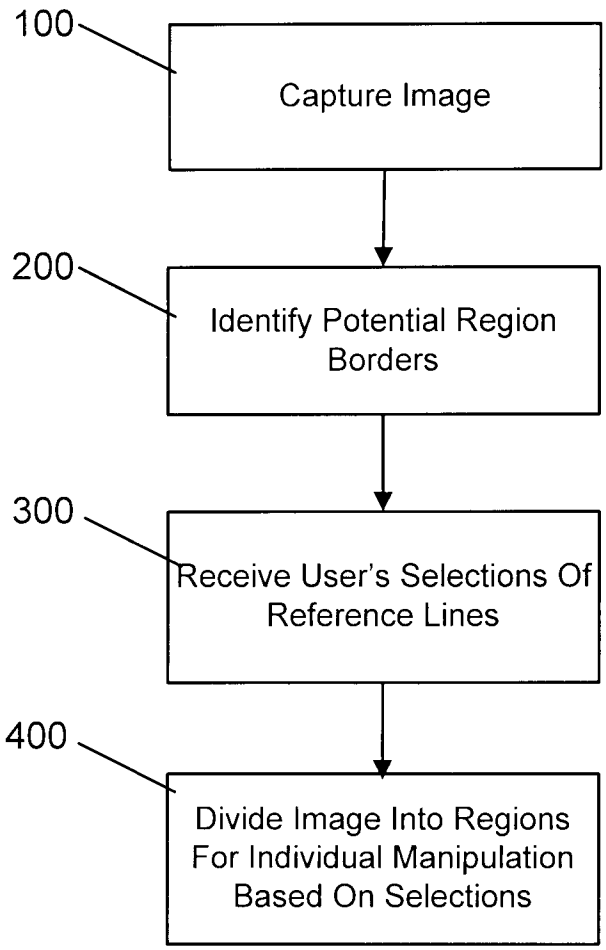

100
Capture Image
200
Identify Potential Region Borders
300
Receive User's Selections Of Reference Lines
400
Divide Image Into Regions For Individual Manipulation Based On Selections

Fig. 1

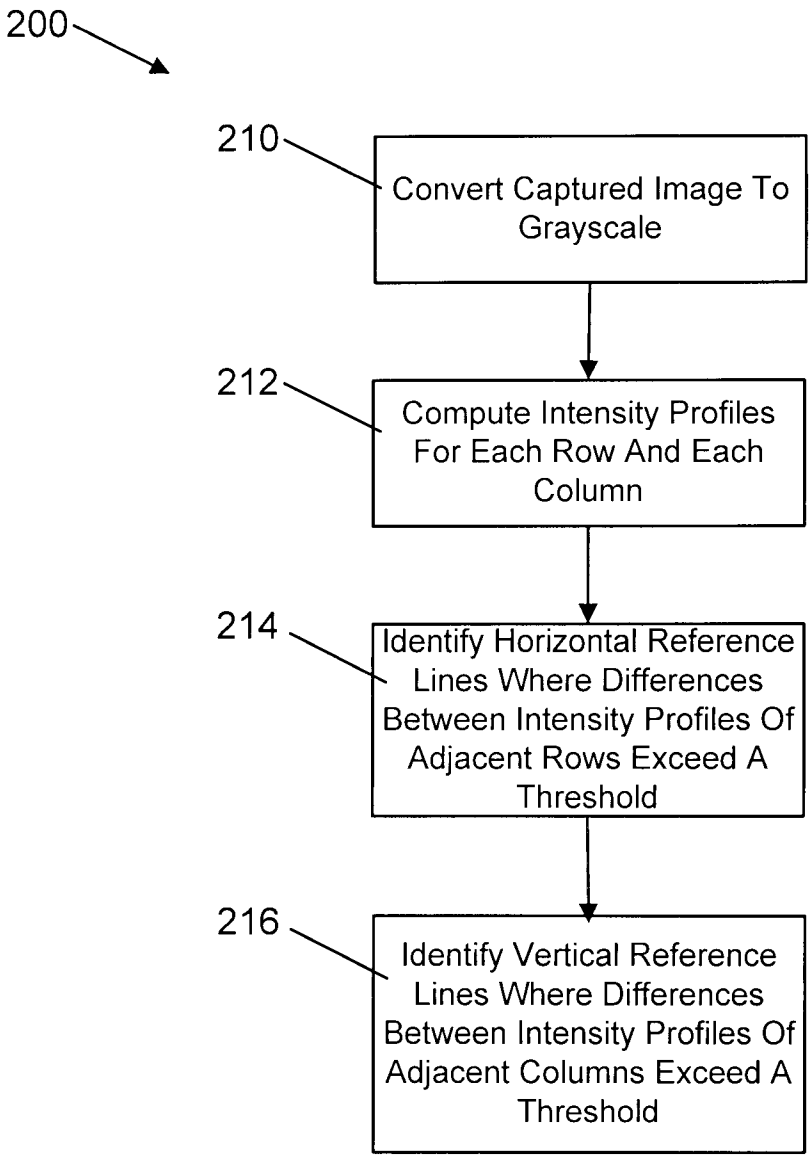

200
210
Convert Captured Image To Grayscale
212
Compute Intensity Profiles For Each Row And Each Column
214
Identify Horizontal Reference Lines Where Differences Between Intensity Profiles Of Adjacent Rows Exceed A Threshold
216
Identify Vertical Reference Lines Where Differences Between Intensity Profiles Of Adjacent Columns Exceed A Threshold

Fig. 2


cnet NEWS.com
Log in | Sign up
Why join?
Today on CNET
News
Reviews
Compare prices
How-to
Downloads
Today on News | Business Tech | Cutting Edge | Access | Threats | Media 2.0 | Markets | Digital Life
My News | Most Popular | Extra | Blogs | Corrections
Search:
Go!
Options
Microsoft releases Windows CE 6 beta
By Aaron Tan
Special to CNET News.com
Published: May 10, 2006, 7:26 AM PDT
TalkBack
E-mail
Print
Microsoft has unveiled a test version of its Windows CE 6 operating system for embedded devices.
The operating system is commonly used in set-top boxes, GPS (Global Positioning System) devices, PDAs, industrial automation and medical devices. Products with the new OS are expected next year.
Microsoft said it has redesigned the kernel, or heart of the OS, and exponentially expanded the capacity for simultaneous processes from 32 to 32,000.
This means developers can incorporate large numbers of complex applications into devices and offer advanced multimedia and Web services. Windows CE 6 does include features from older iterations of Windows CE, allowing device makers to keep previous investments in user interfaces, applications, middleware and drivers.
Microsoft also introduced an embedded development environment available via a plug-in for Visual Studio 2005. The move will encourage some 7 million Visual Studio developers worldwide to program for embedded devices, the company said.
As part of Microsoft's moves to counter open-source rivals like Linux in the embedded space, developers can also access millions of lines of Windows CE source code, under Microsoft's Shared Source
Get Started NOW
AMD
Get More out of Your Datacenter
AMD In the Enterprise | An Online Series
Power and Energy in the Enterprise
Event #5 Happening NOW
E-mail alerts
The headlines you want, straight to your in-box
REGISTER HERE
THE BIG PICTURE
RELATED STORIES
WHAT'S HOT
LATEST HEADLINES
• Yahoo, Telemundo to merge Spanish-language sites

Fig. 3


Column intensity profile
Row intensity profile
cnet NEWS.com
Log in | Sign up
Why join?
Today on CNET
News
Reviews
Compare prices
How-to
Downloads
Today on News | Business Tech | Cutting Edge | Access | Threats | Media 2.0 | Markets | Digital Life
My News | Most Popular | Extra | Blogs | Corrections
Search:
Go!
Options
Microsoft releases Windows CE 6 beta
By Aaron Tan
Special to CNET News.com
Published: May 10, 2006, 7:26 AM PDT
TalkBack
E-mail
Print
Microsoft has unveiled a test version of its Windows CE 6 operating system for embedded devices.
The operating system is commonly used in set-top boxes, GPS (Global Positioning System) devices, PDAs, industrial automation and medical devices. Products with the new OS are expected next year.
Microsoft said it has redesigned the kernel, or heart of the OS, and exponentially expanded the capacity for simultaneous processes from 32 to 32,000.
This means developers can incorporate large numbers of complex applications into devices and offer advanced multimedia and Web services. Windows CE 6 does include features from older iterations of Windows CE, allowing device makers to keep previous investments in user interfaces, applications, middleware and drivers.
Microsoft also introduced an embedded development environment available via a plug-in for Visual Studio 2005. The move will encourage some 7 million Visual Studio developers worldwide to program for embedded devices, the company said.
As part of Microsoft's moves to counter open-source rivals like Linux in the embedded space, developers can also access millions of lines of Windows CE source code under Microsoft's Shared Source
Advertisement
Get Started NOW
AMD
Get More out of Your Datacenter
AMD In the Enterprise | An Online Series
Power and Energy in the Enterprise
Event #5 Happening NOW
E-mail alerts
The headlines you want, straight to your in-box
REGISTER HERE
THE BIG PICTURE
RELATED STORIES
WHAT'S HOT
LATEST HEADLINES
Yahoo, Telemundo to merge Spanish-language sites

Fig. 4


cnet NEWS.com
Log in | Sign up
Why join?
Today on CNET
News
Reviews
Compare prices
How-to
Downloads
Today on News | Business Tech | Cutting Edge | Access | Threats | Media 2.0 | Markets | Digital Life
My News | Most Popular | Extra | Blogs | Corrections
Search:
Go!
Options
Microsoft releases Windows CE 6 beta
By Aaron Tan
Special to CNET News.com
Published: May 10, 2006, 7:26 AM PDT
TalkBack
E-mail
Print
Microsoft has unveiled a test version of its Windows CE 6 operating system for embedded devices.
The operating system is commonly used in set-top boxes, GPS (Global Positioning System) devices, PDAs, industrial automation and medical devices. Products with the new OS are expected next year.
Microsoft said it has redesigned the kernel, or heart of the OS, and exponentially expanded the capacity for simultaneous processes from 32 to 32,000.
This means developers can incorporate large numbers of complex applications into devices and offer advanced multimedia and Web services. Windows CE 6 does include features from older iterations of Windows CE, allowing device makers to keep previous investments in user interfaces, applications, middleware and drivers.
Microsoft also introduced an embedded development environment available via a plug-in for Visual Studio 2005. The move will encourage some 7 million Visual Studio developers worldwide to program for embedded devices, the company said.
As part of Microsoft's moves to counter open-source rivals like Linux in the embedded space, developers can also access millions of lines of Windows CE source code, under Microsoft's Shared Source
advertisement
Get Started NOW
AMD
Get More out of Your Datacenter
AMD In the Enterprise | An Online Series
Power and Energy in the Enterprise
Event #5 Happening NOW
E-mail alerts
The headlines you want, straight to your in-box
REGISTER HERE
THE BIG PICTURE
RELATED STORIES
WHAT'S HOT
LATEST HEADLINES
• Yahoo, Telemundo to merge Spanish-language sites

Fig. 5


c|net NEWS.com
Log in | Sign up
Why join?
Today on CNET
News
Reviews
Compare prices
How-to
Downloads
Today on News | Business Tech | Cutting Edge | Access | Threats | Media 2.0 | Markets | Digital Life
My News | Most Popular | Extra | Blogs | Corrections
Search:
Go!
Options
Microsoft releases Windows CE 6 beta
By Aaron Tan
Special to CNET News.com
Published: May 10, 2006, 7:26 AM PDT
TalkBack
E-mail
Print
Microsoft has unveiled a test version of its Windows CE 6 operating system for embedded devices.
The operating system is commonly used in set-top boxes, GPS (Global Positioning System) devices, PDAs, industrial automation and medical devices. Products with the new OS are expected next year.
Microsoft said it has redesigned the kernel, or heart of the OS, and exponentially expanded the capacity for simultaneous processes from 32 to 32,000.
This means developers can incorporate large numbers of complex applications into devices and offer advanced multimedia and Web services. Windows CE 6 does include features from older iterations of Windows CE, allowing device makers to keep previous investments in user interfaces, applications, middleware and drivers.
Microsoft also introduced an embedded development environment available via a plug-in for Visual Studio 2005. The move will encourage some 7 million Visual Studio developers worldwide to program for embedded devices, the company said.
As part of Microsoft's moves to counter open-source rivals like Linux in the embedded space, developers can also access millions of lines of Windows CE source code, under Microsoft's Shared Source
Get Started NOW
AMD
Get More out of Your Datacenter
AMD In the Enterprise | An Online Series
Power and Energy in the Enterprise
Event #5 Happening NOW
E-mail alerts
The headlines you want, straight to your in-box
REGISTER HERE
THE BIG PICTURE
RELATED STORIES
WHAT'S HOT
LATEST HEADLINES
• Yahoo, Telemundo to merge Spanish-language sites

Fig. 6

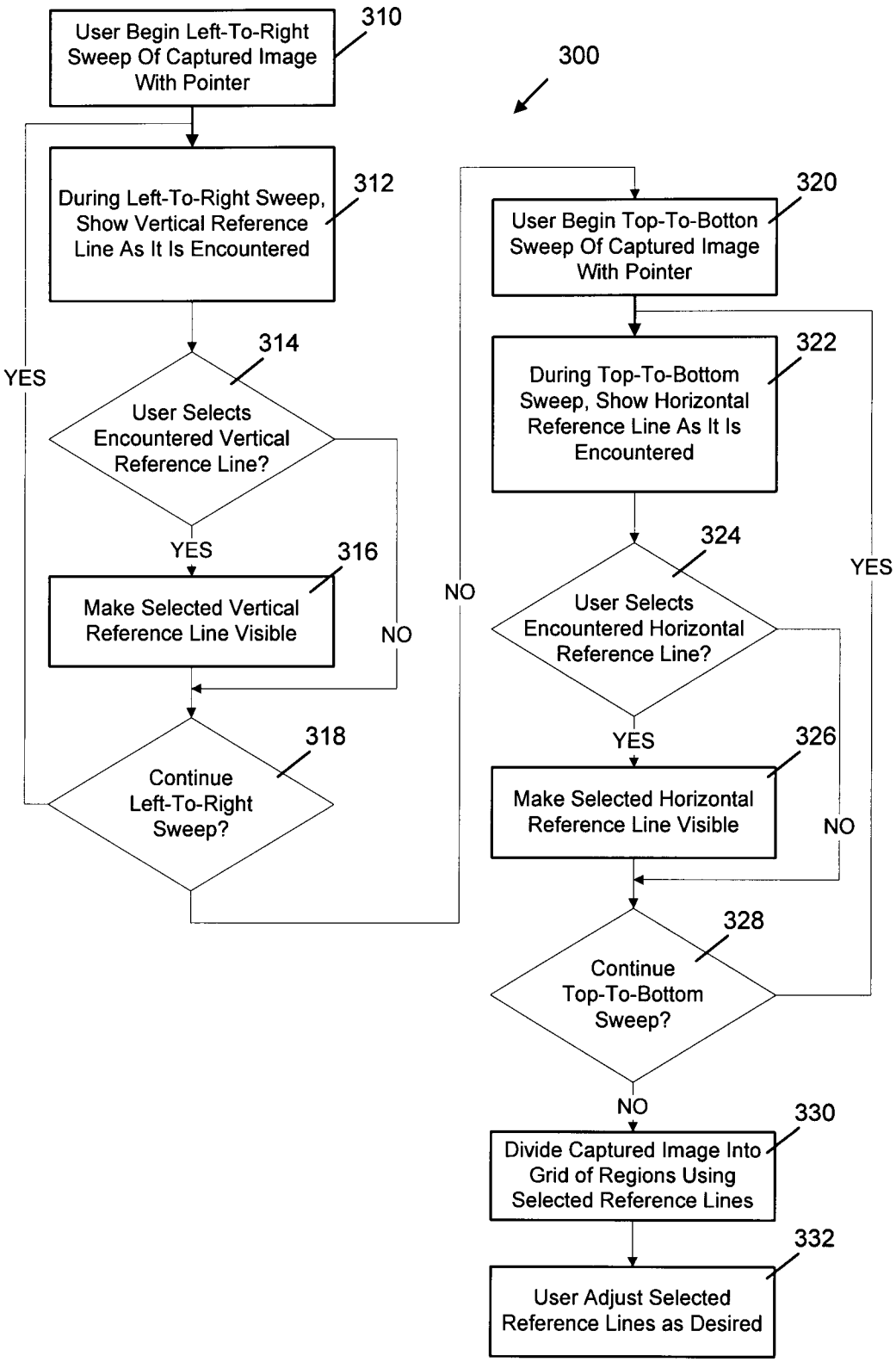

300
User Begin Left-To-Right Sweep Of Captured Image With Pointer
310
During Left-To-Right Sweep, Show Vertical Reference Line As It Is Encountered
312
User Selects Encountered Vertical Reference Line?
314
YES
NO
Make Selected Vertical Reference Line Visible
316
Continue Left-To-Right Sweep?
318
User Begin Top-To-Botton Sweep Of Captured Image With Pointer
320
During Top-To-Bottom Sweep, Show Horizontal Reference Line As It Is Encountered
322
User Selects Encountered Horizontal Reference Line?
324
YES
NO
Make Selected Horizontal Reference Line Visible
326
Continue Top-To-Bottom Sweep?
328
YES
NO
Divide Captured Image Into Grid of Regions Using Selected Reference Lines
330
User Adjust Selected Reference Lines as Desired
332

Fig. 7


cnet NEWS.com
Log in | Sign up
Why join?
Today on CNET
News
Reviews
Compare prices
How-to
Downloads
Today on News | Business Tech | Cutting Edge | Access | Threats | Media 2.0 | Markets | Digital Life
My News | Most Popular | Extra | Blogs | Corrections
Search:
Go!
Options
Microsoft releases Windows CE 6 beta
By Aaron Tan
Special to CNET News.com
Published: May 10, 2006, 7:26 AM PDT
TalkBack
E-mail
Print
Microsoft has unveiled a test version of its Windows CE 6 operating system for embedded devices.
The operating system is commonly used in set-top boxes, GPS (Global Positioning System) devices, PDAs, industrial automation and medical devices. Products with the new OS are expected next year.
Microsoft said it has redesigned the kernel, or heart of the OS, and exponentially expanded the capacity for simultaneous processes from 32 to 32,000.
This means developers can incorporate large numbers of complex applications into devices and offer advanced multimedia and Web services. Windows CE 6 does include features from older iterations of Windows CE, allowing device makers to keep previous investments in user interfaces, applications, middleware and drivers.
Microsoft also introduced an embedded development environment available via a plug-in for Visual Studio 2005. The move will encourage some 7 million Visual Studio developers worldwide to program for embedded devices, the company said.
As part of Microsoft's moves to counter open-source rivals like Linux in the embedded space, developers can also access millions of lines of Windows CE source code, under Microsoft's Shared Source
advertisement
Get Started NOW
AMD
Get More out of Your Datacenter
AMD In the Enterprise | An Online Series
Power and Energy in the Enterprise
Event #5 Happening NOW
E-mail alerts
The headlines you want, straight to your in-box
REGISTER HERE
THE BIG PICTURE
RELATED STORIES
WHAT'S HOT
LATEST HEADLINES
• Yahoo, Telemundo to merge Spanish-language sites

Fig. 8


cnet NEWS.com
Log in | Sign up
Why join?
Today on CNET
News
Reviews
Compare prices
How-to
Downloads
Today on News | Business Tech | Cutting Edge | Access | Threats | Media 2.0 | Markets | Digital Life
My News | Most Popular | Extra | Blogs | Corrections
Search:
Go!
Options
Microsoft releases Windows CE 6 beta
By Aaron Tan
Special to CNET News.com
Published: May 10, 2006, 7:26 AM PDT
TalkBack
E-mail
Print
Microsoft has unveiled a test version of its Windows CE 6 operating system for embedded devices.
The operating system is commonly used in set-top boxes, GPS (Global Positioning System) devices, PDAs, industrial automation and medical devices. Products with the new OS are expected next year.
Microsoft said it has redesigned the kernel, or heart of the OS, and exponentially expanded the capacity for simultaneous processes from 32 to 32,000.
This means developers can incorporate large numbers of complex applications into devices and offer advanced multimedia and Web services. Windows CE 6 does include features from older iterations of Windows CE, allowing device makers to keep previous investments in user interfaces, applications, middleware and drivers.
Microsoft also introduced an embedded development environment available via a plug-in for Visual Studio 2005. The move will encourage some 7 million Visual Studio developers worldwide to program for embedded devices, the company said.
As part of Microsoft's moves to counter open-source rivals like Linux in the embedded space, developers can also access millions of lines of Windows CE source code, under Microsoft's Shared Source
advertisement
Get Started NOW
AMD
Get More out of Your Datacenter
AMD In the Enterprise | An Online Series
Power and Energy in the Enterprise
Event #5 Happening NOW
E-mail alerts
The headlines you want, straight to your in-box
REGISTER HERE
THE BIG PICTURE
RELATED STORIES
WHAT'S HOT
LATEST HEADLINES
Yahoo, Telemundo to merge Spanish-language sites

Fig. 9

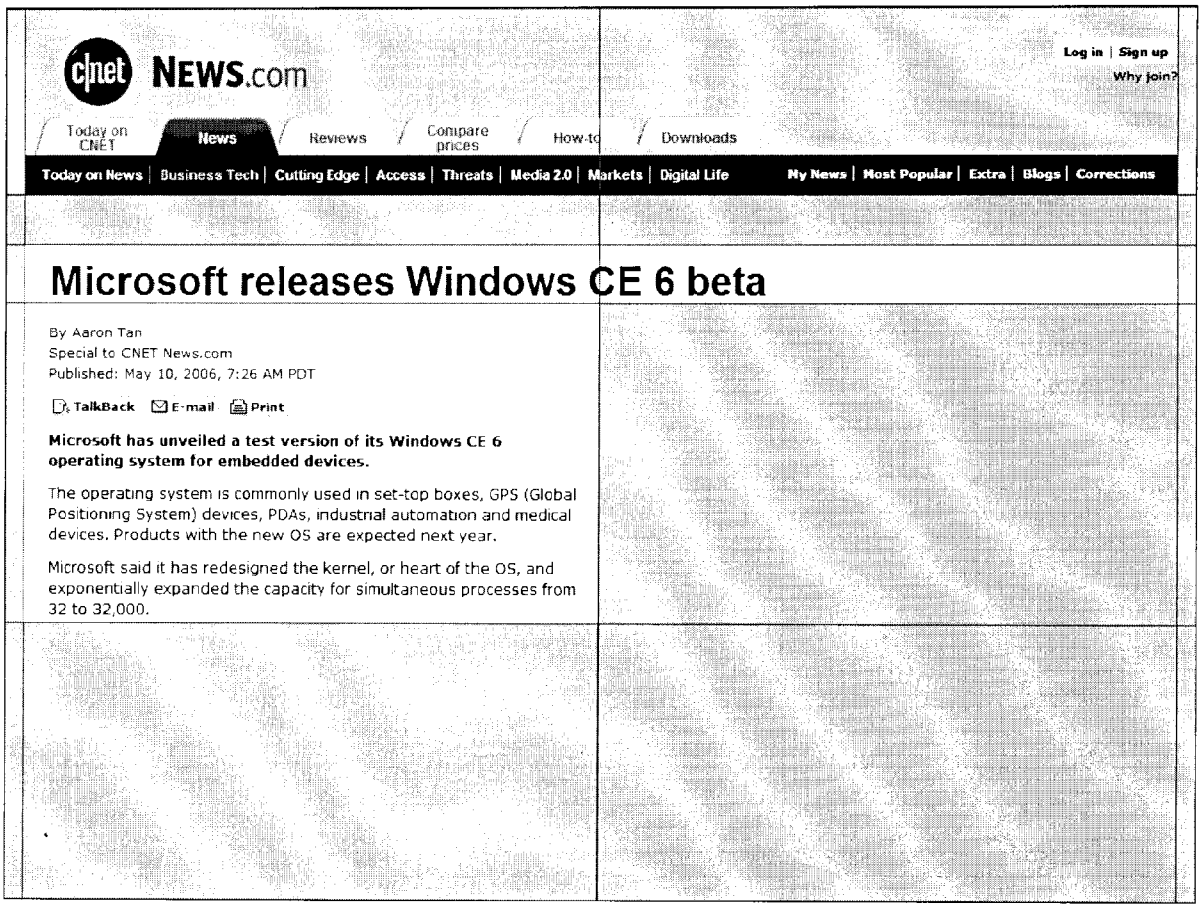

cnet NEWS.com
Log in | Sign up
Why join?
Today on CNET
News
Reviews
Compare prices
How-to
Downloads
Today on News | Business Tech | Cutting Edge | Access | Threats | Media 2.0 | Markets | Digital Life
My News | Most Popular | Extra | Blogs | Corrections
Microsoft releases Windows CE 6 beta
By Aaron Tan
Special to CNET News.com
Published: May 10, 2006, 7:26 AM PDT
TalkBack E-mail Print
Microsoft has unveiled a test version of its Windows CE 6 operating system for embedded devices.
The operating system is commonly used in set-top boxes, GPS (Global Positioning System) devices, PDAs, industrial automation and medical devices. Products with the new OS are expected next year.
Microsoft said it has redesigned the kernel, or heart of the OS, and exponentially expanded the capacity for simultaneous processes from 32 to 32,000.

Fig. 10

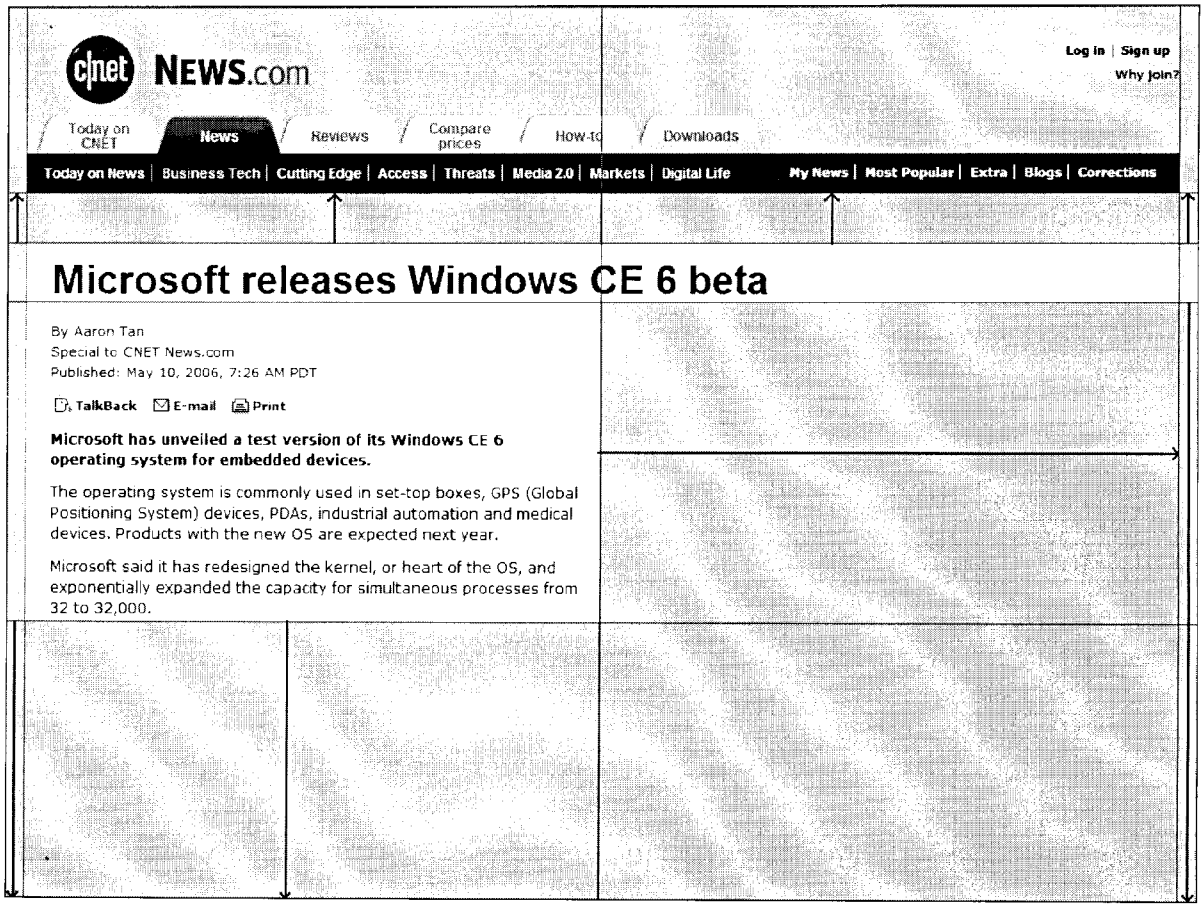

cnet NEWS.com
Log in | Sign up
Why join?
Today on CNET
News
Reviews
Compare prices
How-to
Downloads
Today on News | Business Tech | Cutting Edge | Access | Threats | Media 2.0 | Markets | Digital Life
My News | Most Popular | Extra | Blogs | Corrections
Microsoft releases Windows CE 6 beta
By Aaron Tan
Special to CNET News.com
Published: May 10, 2006, 7:26 AM PDT
TalkBack
E-mail
Print
Microsoft has unveiled a test version of its Windows CE 6 operating system for embedded devices.
The operating system is commonly used in set-top boxes, GPS (Global Positioning System) devices, PDAs, industrial automation and medical devices. Products with the new OS are expected next year.
Microsoft said it has redesigned the kernel, or heart of the OS, and exponentially expanded the capacity for simultaneous processes from 32 to 32,000.

Fig. 11

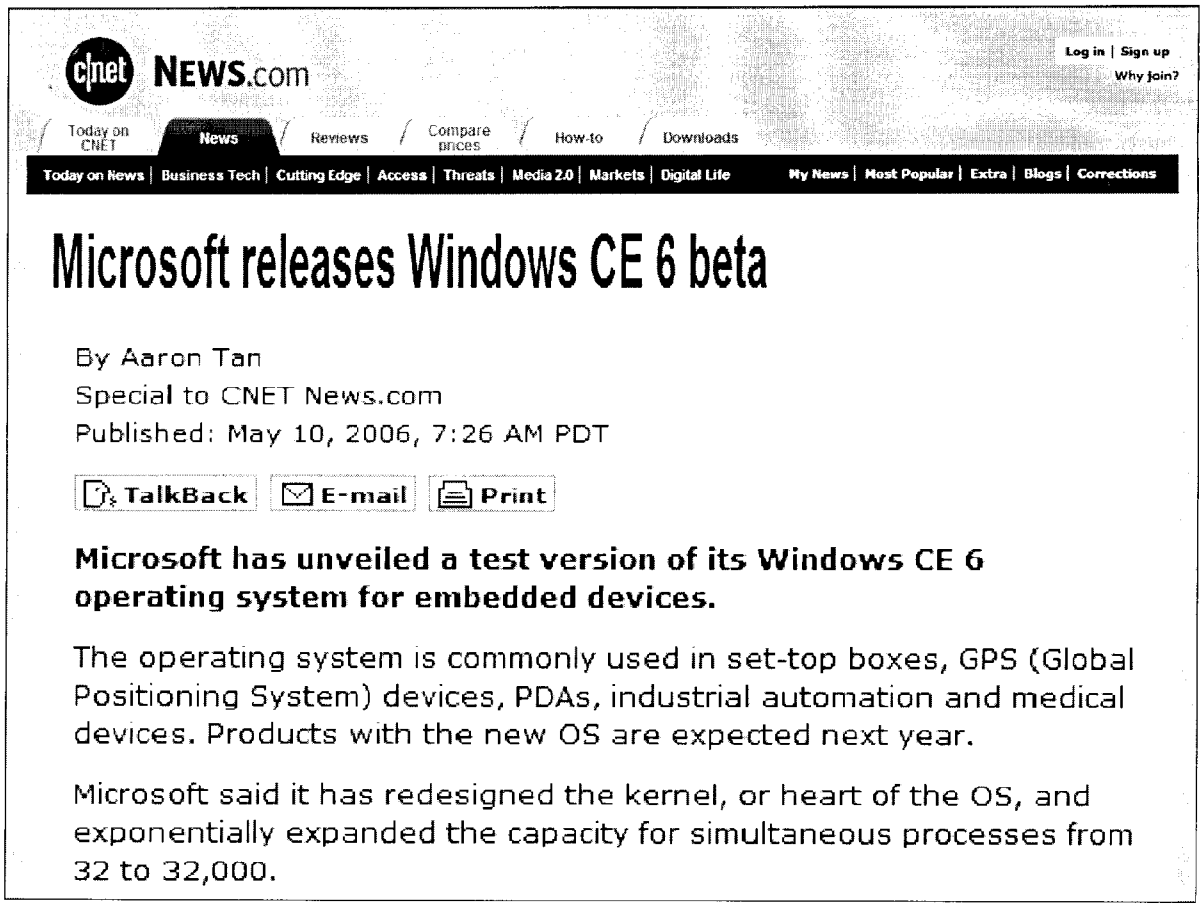

cnet NEWS.com
Log in | Sign up
Why join?
Today on CNET
News
Reviews
Compare prices
How-to
Downloads
Today on News | Business Tech | Cutting Edge | Access | Threats | Media 2.0 | Markets | Digital Life
My News | Most Popular | Extra | Blogs | Corrections
Microsoft releases Windows CE 6 beta
By Aaron Tan
Special to CNET News.com
Published: May 10, 2006, 7:26 AM PDT
TalkBack
E-mail
Print
Microsoft has unveiled a test version of its Windows CE 6 operating system for embedded devices.
The operating system is commonly used in set-top boxes, GPS (Global Positioning System) devices, PDAs, industrial automation and medical devices. Products with the new OS are expected next year.
Microsoft said it has redesigned the kernel, or heart of the OS, and exponentially expanded the capacity for simultaneous processes from 32 to 32,000.

Fig. 12

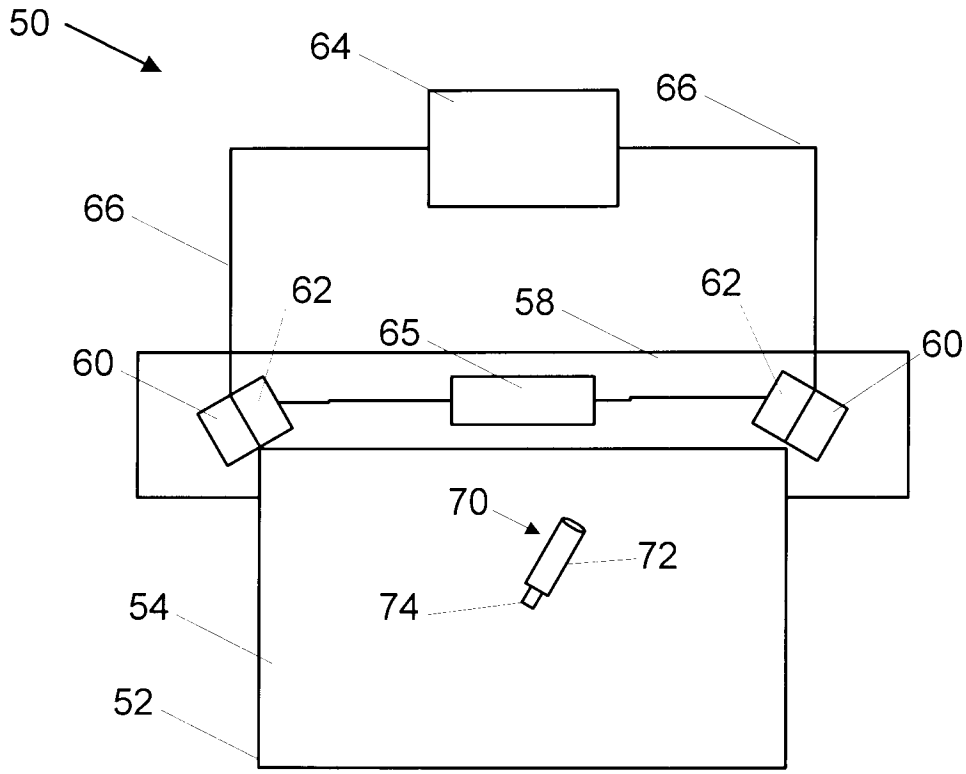



Fig. 13

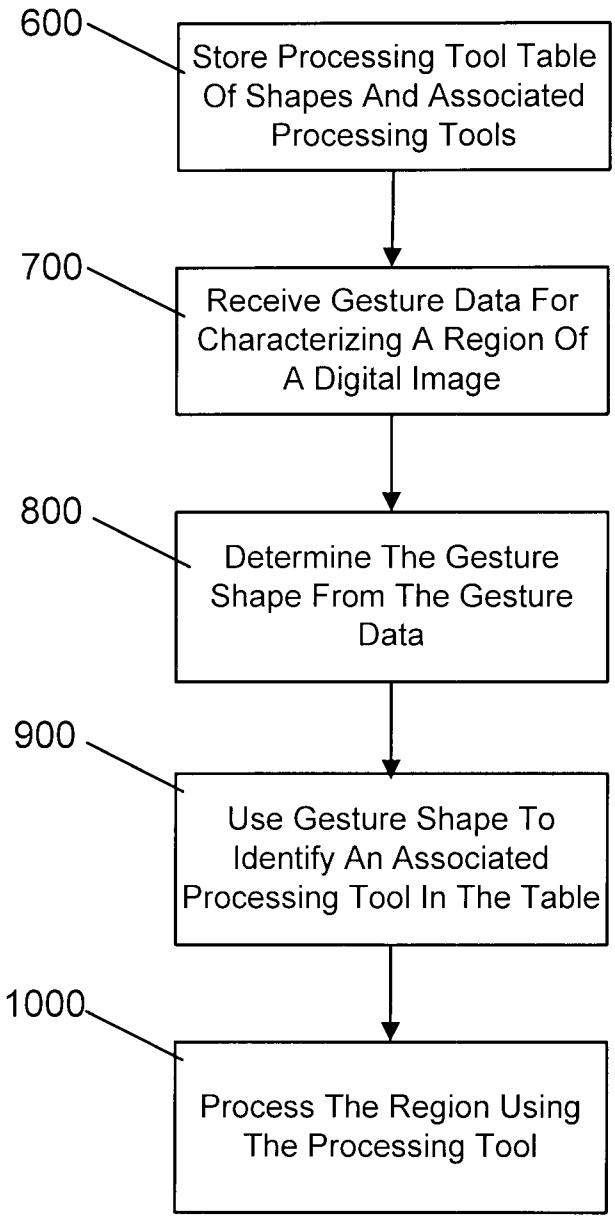

600
Store Processing Tool Table Of Shapes And Associated Processing Tools
700
Receive Gesture Data For Characterizing A Region Of A Digital Image
800
Determine The Gesture Shape From The Gesture Data
900
Use Gesture Shape To Identify An Associated Processing Tool In The Table
1000
Process The Region Using The Processing Tool

Fig. 14




Fig. 15a

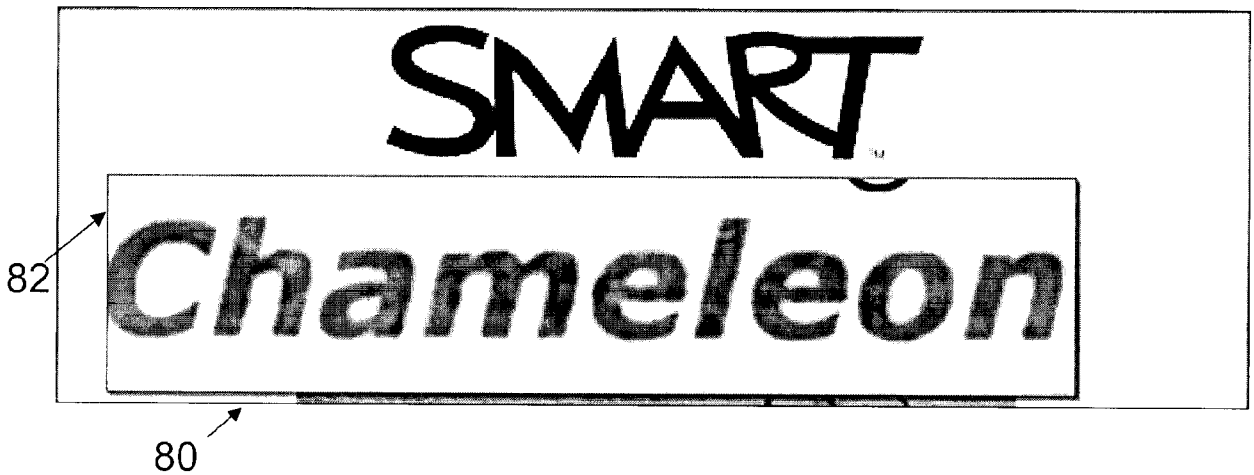



Fig. 15b

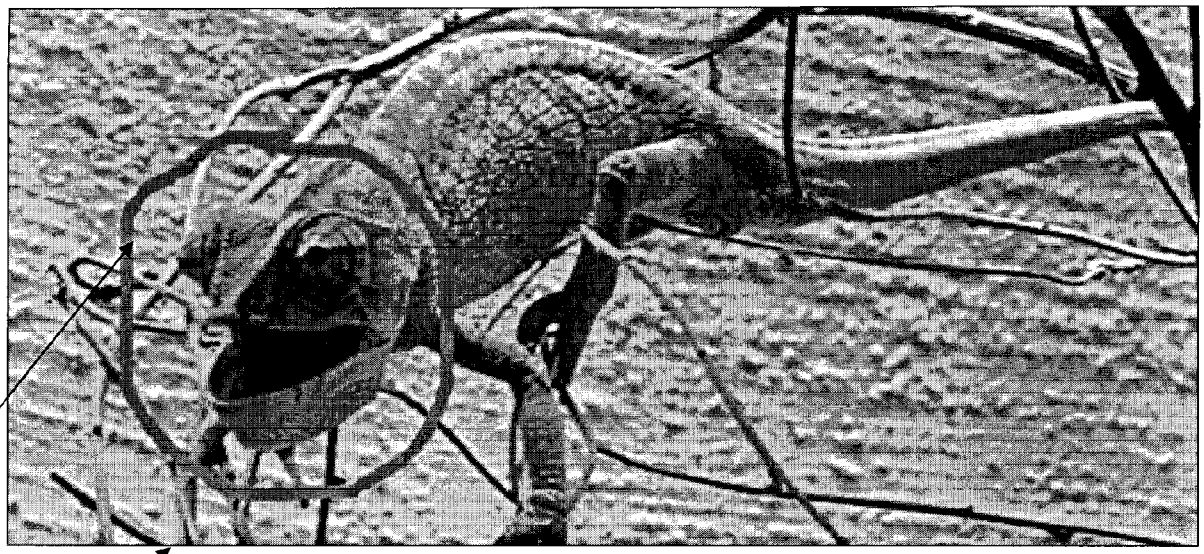



Fig. 16a




Fig. 16b

**INTERNATIONAL SEARCH REPORT**

International application No.
PCT/CA2008/001311

A. CLASSIFICATION OF SUBJECT MATTER
IPC: ***G06T 1/00*** (2006.01), ***G06T 11/60*** (2006.01), ***G06T 3/40*** (2006.01), ***G06T 5/00*** (2006.01), ***G06T 7/60*** (2006.01)

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC: ***G06T 1/00*** (2006.01), ***G06T 11/60*** (2006.01), ***G06T 3/40*** (2006.01), ***G06T 5/00*** (2006.01), ***G06T 7/60*** (2006.01)

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic database(s) consulted during the international search (name of database(s) and, where practicable, search terms used)
**Databases:** Delphion USPTO, Espacenet, Canadian Patent Database, IEEE, Google.
**Keywords:** shape, draw, command, border, image, web site, region, divide, touch screen, gesture, hand, and other related terms.

C. DOCUMENTS CONSIDERED TO BE RELEVANT

| Category* | Citation of document, with indication, where appropriate, of the relevant passages | Relevant to claim No. |
|---|---|---|
| X | US 2006/0274067 A1 (Hidai), 07 December 2006 (07-12-2006). Entire document. | 1, 2, 8, 9, 10 and 27 |
| Y | | 14 to 26 and 28 to 35 |
| Y | US 2005/0052427 A1 (Wu et al.), 10 March 2005 (10-03-2005). Entire document. | 14 to 26 and 28 to 35 |
| A | US 2005/0210399 A1 (Microsoft Corporation), 22 September 2005 (22-09-2005). Entire document. | 1 to 35 |
| A | US 6,345,279 B1 (International Business Machine), 05 February 2002 (05-02-2002). Entire document. | 1 to 35 |

[X] Further documents are listed in the continuation of Box C. [X] See patent family annex.

* Special categories of cited documents :
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

| Date of the actual completion of the international search | Date of mailing of the international search report |
|---|---|
| 17 October 2008 (17-10-2008) | 20 October 2008 (20-10-2008) |

| Name and mailing address of the ISA/CA | Authorized officer |
|---|---|
| Canadian Intellectual Property Office<br>Place du Portage I, C114 - 1st Floor, Box PCT<br>50 Victoria Street<br>Gatineau, Quebec K1A 0C9<br>Facsimile No.: 001-819-953-2476 | Enji Al-Saber 819- 934-5142 |

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

| Category* | Citation of document, with indication, where appropriate, of the relevant passages | Relevant to claim No. |
|---|---|---|
| A | US 6,300,947 B1 (International Business Machine), 09 October 2001 (09-10-2001).<br>Entire document. | 1 to 35 |
| A | US 7,203,901 B2 (Microsoft Corporation), 10 April 2007 (10-04-2007).<br>Entire document. | 1 to 35 |

**INTERNATIONAL SEARCH REPORT**
Information on patent family members

International application No.
PCT/CA2008/001311

| Patent Document Cited in Search Report | Publication Date | Patent Family Member(s) | Publication Date |
|---|---|---|---|
| US2006274067 | 07-12-2006 | JP2003162370 A | 06-06-2003 |
| | | US7184045 B2 | 27-02-2007 |
| | | US7423654 B2 | 09-09-2008 |
| US2005052427 | 10-03-2005 | JP2005100391 A | 14-04-2005 |
| US2005210399 | 22-09-2005 | NONE | |
| US6345279 | 05-02-2002 | NONE | |
| US6300947 | 09-10-2001 | CN1152320C C | 02-06-2004 |
| | | GB2339374 A | 19-01-2000 |
| | | JP3725369B2 B2 | 07-12-2005 |
| | | KR20000011423 A | 25-02-2000 |
| US7203901 | 10-04-2007 | US2006282444 A1 | 14-12-2006 |
| | | US2006282445 A1 | 14-12-2006 |