cases, the virtual scroll wheel may include a virtual button at its center. The virtual scroll wheel is configured to implement scrolling as for example through a list and the button is configured to implement selections as for example items stored in the list. Following block **752**, the method proceeds to block **754** where the presence of at least a first finger and more particularly, first and second fingers (to distinguish between tracking and gesturing) over the virtual scroll wheel is detected on a touch screen. The touch screen is positioned over or in front of the display. By way of example, the display may be an LCD and the touch screen may be a multipoint touch screen. Following block **754**, the method proceeds to block **756** where the initial position of the fingers on the virtual scroll wheel is set. By way of example, the angle of the fingers relative to a reference point may be determined (e.g., 12 o clock, 6 o clock, etc.). Following block **756**, the method **750** proceeds to block **758** where a rotate signal is generated when the angle of the fingers change relative to the reference point. In most cases, the set down of the fingers associate, link or lock the fingers (or finger) to the virtual scroll wheel when the fingers are positioned over the virtual scroll wheel. As a result, when the fingers are rotated, the rotate signal can be used to rotate the virtual scroll wheel in the direction of finger rotation (e.g., clockwise, counterclockwise). In most cases, the amount of wheel rotation varies according to the amount of finger rotation, i.e., if the fingers move 5 degrees then so will the wheel. Furthermore, the rotation typically occurs substantially simultaneously with the motion of the fingers. For instance, as the fingers rotate, the scroll wheel rotates with the fingers at the same time.

[**0125**] In some cases, the principals of inertia as described above can be applied to the virtual scroll wheel. In cases such as these, the virtual scroll wheel continues to rotate when the fingers (or one of the fingers) are lifted off of the virtual scroll wheel and slowly comes to a stop via virtual friction. Alternatively or additionally, the continuous rotation can be stopped by placing the fingers (or the removed finger) back on the scroll wheel thereby braking the rotation of the virtual scroll wheel.

[**0126**] **FIGS. 27A-27D** illustrates a scroll wheel sequence using the method described above. **FIG. 27A** illustrates a display presenting a scroll wheel. The scroll wheel may be displayed automatically as part of a program or it may be displayed when a particular gesture is performed. By way of example, during the operation of a music program (such as iTunes manufactured by Apple Computer Inc., of Cupertino, Calif.), the virtual scroll wheel may appear on the GUI of the music program when two fingers are placed on the touch screen rather than one finger which is typically used for tracking in the music program. In some cases, the virtual scroll wheel only appears when two fingers are placed on a predetermined area of the GUI. As shown in **FIG. 27B**, a user positions their fingers over the multipoint touch screen **520** over the scroll wheel. At some point, the fingers are locked to the scroll wheel. This can occur at set down for example. As shown in **FIG. 27C**, when the fingers are rotated in a clockwise direction, the scroll wheel is rotated in the clockwise direction in accordance with the rotating fingers. As shown in **FIG. 27D**, when the fingers are rotated in a counterclockwise direction, the virtual scroll wheel is rotated in the counter clockwise direction in accordance with the rotating fingers. Alternatively, rotation of the virtual scroll wheel may also be rotated with linear motion of the fingers in a tangential manner.

[**0127**] It should be noted that although a surface scroll wheel is shown, the principals thereof can be applied to more conventional scroll wheels which are virtually based. For example, scroll wheels, whose axis is parallel to the display screen and which appear to protrude through the display screen as shown in **FIG. 28**. In this particular implementation, however, linear motion of the fingers are used to rotate the virtual scroll wheel.

[**0128**] The various aspects, embodiments, implementations or features of the invention can be used separately or in any combination.

[**0129**] The invention is preferably implemented by hardware, software or a combination of hardware and software. The software can also be embodied as computer readable code 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 the computer readable medium include read-only memory, random-access memory, CD-ROMs, DVDs, magnetic tape, optical data storage devices, and carrier waves. The computer readable medium can also be distributed over network-coupled computer systems so that the computer readable code is stored and executed in a distributed fashion.

[**0130**] While this invention has been described in terms of several preferred embodiments, there are alterations, permutations, and equivalents, which fall within the scope of this invention. For example, although the invention has been primarily directed at touchscreens, it should be noted that in some cases touch pads may also be used in place of touchscreens. Other types of touch sensing devices may also be utilized. It should also be noted that there are many alternative ways of implementing the methods and apparatuses of the present invention. It is therefore intended that the following appended claims be interpreted as including all such alterations, permutations, and equivalents as fall within the true spirit and scope of the present invention.

What is claimed is:

1. A computer implemented method for processing touch inputs, said method comprising:

reading data from a touch sensitive device, the data pertaining to touch input with respect to the touch sensitive device, and the touch sensitive device having a multipoint capability; and

identifying at least one multipoint gesture based on the data from the touch sensitive device.

2. A method of invoking a user interface element on a display via a multipoint touch sensitive device of a computing system, said method comprising:

detecting and analyzing the simultaneous presence of two or more objects in contact with said touch sensitive device;

based at least in part on said analyzing, selecting a user interface tool, from a plurality of available tools, to display on a display for interaction by a user of said computing system; and