and will eventually be deleted through either tap timeout **782** when the finger lifts off or through tap timeout **796**) if another touches down after it.

[0293] Step **804** simply sets a flag which will indicate to decision diamond **798** during future scan cycles that typematic has already started for the element. Upon typematic initialization, step **810** sends out the key symbol for the first time to the host interface communication queue, along with any modifier symbols being held down by the opposite hand. Step **812** records the time the key symbol is sent for future reference by decision diamond **808**. Processing then returns to step **770** to await the next proximity image scan.

[0294] Until the finger lifts off or another taps asynchronously, processing will pass through decision diamond **798** to check whether the key symbol should be sent again. Step **806** computes the symbol repeat interval dynamically to be inversely proportional to finger proximity. Thus the key will repeat faster as the finger is pressed on the surface harder or a larger part of the fingertip touches the surface. This also reduces the chance that the user will cause more repeats than intended since as finger proximity begins to drop during liftoff the repeat interval becomes much longer. Decision diamond **808** checks whether the dynamic repeat interval since the last typematic symbol send has elapsed, and if necessary sends the symbol again in **810** and updates the typematic send time stamp **812**.

[0295] It is desirable to let the users rest the other fingers back onto the surface after typematic has initiated **804** and while typematic continues, but the user must do so without tapping. Decision diamond **805** causes typematic to be canceled and the typematic element deleted **778** if the user asynchronously taps another finger on the surface as if trying to hit another key. If this does not occur, decision diamond **182** will eventually cause deletion of the typematic element when its finger lifts off.

[0296] The typing recognition process described above thus allows the multi-touch surface to ergonomically emulate both the typing and hand resting capabilities of a standard mechanical keyboard. Crisp taps or impulsive presses on the surface generate key symbols as soon as the finger is released or decision diamond **792** verifies the impulse has peaked, ensuring prompt feedback to the user. Fingers intended to rest on the surface generate no keys as long as they are members of a synchronized finger press or release subset or are placed on the surface gently and remain there along with other fingers for a second or two. Once resting, fingers can be lifted and tapped or impulsively pressed on the surface to generate key symbols without having to lift other resting fingers. Typematic is initiated ether by impulsively pressing and maintaining distinguishable force on a key, or by holding a finger on a key while other fingers on the hand are lifted. Glancing motions of single fingers as they tap key regions are easily tolerated since most cursor manipulation must be initiated by synchronized slides of two or more fingers.

[0297] Other embodiments of the invention will be apparent to those skilled in the art from consideration of the specification and practice of the invention disclosed herein. It is intended that the specification and examples be considered as exemplary only, with a true scope and spirit of the invention being indicated by the following claims.

What is claimed is:

1. A method for extracting multiple degrees of freedom of hand motion from successive proximity images, the method comprising:

tracking a plurality of contacts associated with a plurality of hand parts across the successive proximity images;

finding an innermost finger contact and an outermost finger contact for a given hand from the plurality of contacts;

computing a scaling velocity component from a change in a distance between the innermost and outermost finger contacts; and

transmitting the computed scaling velocity component as a control signal to an electronic or electromechanical device.

2. The method of claim 1 further comprising:

computing a rotational velocity component from a change in a vector angle between the innermost and outermost finger contacts; and

transmitting the computed rotational velocity component as a control signal to an electronic or electromechanical device.

3. The method of claim 2 further comprising:

computing a translation weighting for each contact associated with a finger;

computing translational velocity components for each contact associated with a finger;

computing a translational velocity average from the computed translational velocity components and the computed translation weightings; and

transmitting the computed translational velocity average as a control signal to an electronic or electromechanical device.

4. The method of claim 1 further comprising:

computing a translation weighting for each contact associated with a finger;

computing translational velocity components for each contact associated with a finger;

computing a translational velocity average from the computed translational velocity components and the computed translation weightings; and

transmitting the computed translational velocity average as a control signal to an electronic or electromechanical device.

5. A method for extracting multiple degrees of freedom of hand motion from successive proximity images, the method comprising:

tracking a plurality of contacts associated with a plurality of hand parts across the successive proximity images;

finding an innermost finger contact and an outermost finger contact for a given hand from the plurality of contacts;

computing a rotational velocity component from a change in a vector angle between the innermost and outermost finger contacts; and