[0086] FIG. **32** is a plot of the inner finger angle factor versus the angle between the innermost and next innermost finger contacts;

[0087] FIG. **33** is a plot of the inter-hand separation factor versus the estimated distance between the right thumb and left thumb;

[0088] FIG. **34** is a flow chart of hand motion component extraction;

[0089] FIG. **35** is a diagram of typical finger trajectories when hand is contracting;

[0090] FIG. **36** is a flow chart of radial and angular hand velocity extraction;

[0091] FIG. **37** is a flow chart showing extraction of translational hand velocity components;

[0092] FIG. **38** is a flow chart of differential hand pressure extraction;

[0093] FIG. **39**A is a flow chart of the finger synchronization detection loop;

[0094] FIG. **39**B is a flow chart of chord tap detection;

[0095] FIG. **40**A is a flow chart of the chord motion recognition loop;

[0096] FIG. **40**B is a flow chart of chord motion event generation;

[0097] FIG. **41** is a flow chart of key layout morphing;

[0098] FIG. **42** is a flow chart of the keypress detection loop;

[0099] FIG. **43**A is a flow chart of the keypress acceptance and transmission loop; and

[0100] FIG. **43**B is a flow chart of typematic emulation.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0101] Reference will now be made in detail to the present preferred embodiments of the invention, examples of which are illustrated in the accompanying drawings. Wherever possible the same reference numbers will be used throughout the drawings to refer to the same or like parts.

[0102] FIG. **1** is a system block diagram of the entire, integrated manual input apparatus. Sensor embedded in the multi-touch surface **2** detect proximity of entire flattened hands **4**, fingertips thumbs, palms, and other conductive touch devices to the surface **2**. In a preferred embodiment, the surface is large enough to comfortably accommodate both hands **4** and is arched to reduce forearm pronation.

[0103] In alternative embodiments the multi-touch surface **2** may be large enough to accommodate motion of one hand, but may be flexible so it can be fitted to an armrest or clothing.

[0104] Electronic scanning hardware **6** controls and reads from each proximity sensor of a sensor array. A calibration module **8** constructs a raw proximity image from a complete scan of the sensor array and subtracts off any background sensor offsets. The background sensor offsets can simply be a proximity image taken when nothing is touching the surface.

[0105] The offset-corrected proximity image is then passed on to the contact tracking and identification module **10**, which segments the image into distinguishable hand-surface contacts, tracks and identifies them as they move through successive images.

[0106] The paths of identified contacts are passed on to a typing recognizer module **12**, finger synchronization detection module **14**, motion component extraction module **16**, and pen grip detection module **17**, which contain software algorithms to distinguish hand configurations and respond to detected hand motions.

[0107] The typing recognizer module **12** responds to quick presses and releases of fingers which are largely asynchronous with respect to the activity of other fingers on the same hand. It attempts to find the key region nearest to the location of each finger tap and forwards the key symbols or commands associated with the nearest key region to the communication interface module **20**.

[0108] The finger synchronization detector **14** checks the finger activity within a hand for simultaneous presses or releases of a subset of fingers. When such simultaneous activity is detected it signals the typing recognizer to ignore or cancel keystroke processing for fingers contained in the synchronous subset. It also passes on the combination of finger identities in the synchronous subset to the chord motion recognizer **18**.

[0109] The motion component extraction module **16** computes multiple degrees of freedom of control from individual finger motions during easily performable hand manipulations on the surface **2**, such as hand translations, hand rotation about the wrist, hand scaling by grasping with the fingers, and differential hand tilting.

[0110] The chord motion recognizer produces chord tap or motion events dependent upon both the synchronized finger subset identified by the synchronization detector **14** and on the direction and speed of motion extracted in **16**. These events are then posted to the host communication interface **20**.

[0111] The pen grip detection module **17** checks for specific arrangements of identified hand contacts which indicate the hand is configured as if gripping a pen. If such an arrangement is, detected, it forwards the movements of the gripping fingers as inking events to the host communication interface **20**. These inking events can either lay digital ink on the host computer display for drawing or signature capture purposes, or they can be further interpreted by handwriting recognition software which is well known in the art. The detailed steps within each of the above modules will be further described later.

[0112] The host communication interface keeps events from both the typing recognizer **12** and chord motion recognizer **18** in a single temporally ordered queue and dispatches them to the host computer system **22**. The method of communication between the interface **20** and host computer system **22** can vary widely depending on the function and processing power of the host computer. In a preferred embodiment, the communication would take place over computer cables via industry standard protocols such as Apple Desktop Bus, PS/2 keyboard and mouse protocol for PCs, or Universal Serial Bus (USB). In alternative embodiments the software processing of modules **10**-**18** would be performed within the host computer **22**. The multi-touch surface apparatus would only contain enough hardware to scan the proximity sensor array **6**, form proximity images **8**, and compress and send them to the host computer over a wireless network. The host communication interface **20** would then play the role of device driver on the host computer, conveying results of the proximity image recognition process as input to other applications residing on the host computer system **22**.

[0113] In a preferred embodiment the host computer system outputs to a visual display device **24** so that the hands and fingers **4** can manipulate graphical objects on the display screen. However, in alternative embodiments the host computer might output to an audio display or control a machine such as a robot.