

***New Renaissance Institute***<sup>®</sup>

*Technology White Paper*

# **THE NRI<sup>®</sup> RICH TOUCHPAD**

## **Technology and Implementations**

New Renaissance Institute<sup>®</sup>  
P.O. Box 128  
Belmont, California 94002  
[www.newrenaissanceinstitute.com](http://www.newrenaissanceinstitute.com)  
Email: [inquiries@newrenaissanceinstitute.com](mailto:inquiries@newrenaissanceinstitute.com)

## ABSTRACT

*This document describes the technology for, and implementations of, the NRI<sup>®</sup> rich touchpad, a novel touchpad controller, based on U.S. Patent 6,570,078, which can be used for a wide variety of real-time applications. The touchpad is very powerful, provides an unprecedented capability to enter large amounts of information at high speed and is extremely flexible in the kinds of input it can process, the kinds of output it can produce and in how it is configured. The touchpad creates images of the pressure exerted on it and can recognize images created by contact with different parts of the hand (e.g. a fingertip, flat finger, palm, wrist). It can process multiple areas of contact simultaneously and can extract the values of a large number (typically three to six) of continuous parameters from each area of contact. The touchpad is simple to use, and its capabilities can easily be extended. It can be favorably compared to conventional computer pointing devices, such as the mouse, trackball and conventional touchpad, which typically provide control of only two continuous parameters at any one time and can process only a single region of contact.*

*The general-purpose nature of the touchpad permits the same basic system to be used in a wide range of applications. They include CAD/CAE workstation control, real-time machine control, human-machine interfaces for the physically disabled and electronic musical instruments. The touchpad can also be used in intelligent machine sensing and robotics applications.*

*The touchpad can be implemented in a variety of ways. In one implementation, it incorporates a two-dimensional pressure-sensor array, a data acquisition and compression stage, an image processing and recognition stage, and an application interface. Special hardware and algorithms permit the data processing and image processing to be carried out in real time. The system can be modularized to support partitions of the sensor array into functionally discrete regions and aggregations of sensor arrays to form larger arrays.*

*This document is based on U.S. Patent 6,570,078 and related issued and pending patents, all licensable from NRI<sup>®</sup>. Detailed hardware and software reference designs can be discussed under negotiable terms. All financial or in-kind proceeds from such arrangements will be used to fund pure academic research at NRI<sup>®</sup>. Contact inquiries@newrenaissance-institute.com for more information.*

## Table of Contents

- 1 Introduction
- 2 Illustrations of User-Level Operation
  - Controlling Six Parameters with One Finger
  - Compound Images
- 3 System Architecture
  - Physical Formats
  - Pressure-Sensor Array
  - Information Flow
  - Data Acquisition and Data Processing
  - Data Processing for a Single Region of Contact
  - Data Processing for Multiple Regions of Contact
  - Image Recognition
  - Application Interface
  - Immunity to Sensor Failures
- 4 Modularity and Alternative Implementations
  - Segmented Busses and Pipelined-Interleaved Scanning
  - Ganging Sensor Arrays
  - User-Level Partitioning
  - Visual Displays
  - Chip Implementations
  - Fabric Implementations
  - Alternatives to Pressure Sensors
  - A Low-Cost, Limited-Capability Touchpad
- 5 Further Information
- References

# 1 Introduction

This document, based on U.S. Patent 6,570,078 [1], describes the technology for, and implementations of, the NRI<sup>®</sup> rich touchpad, a novel controller that can be used for a wide range of data entry and real-time applications. The touchpad was originally envisioned as a human-machine interface, though it can also be used in machine sensing and robotics. When used by the hand, the touchpad provides an unprecedented capability to enter large amounts of information at high speed.

The touchpad incorporates a pressure-sensor array for gathering information in the form of real-time images of the pressure exerted on it [2]. When used as a human-machine interface, these images are typically produced by contact with the user's hand, though they can be produced using other parts of the body, such as the foot. The pressure images are presented to a data acquisition and compression stage, whose output is sent to an image processing and recognition stage. The image processing and recognition stage is used to identify the shape of particular types of pressure images (e.g. images of a fingertip, flat finger, thumb, palm, wrist). The touchpad can process multiple regions of contact simultaneously and can extract the values of a large number (typically three to six) of continuous parameters from each region of contact. An application interface assigns these values to control signals, which can be used to control arbitrary external systems. Special hardware and algorithms enable the data acquisition, the image processing and recognition, and the derivation of parameter values to be carried out in real time [2,3].

The rich touchpad can be favorably compared to conventional computer pointing devices, such as the mouse, trackball, stylus tablet and conventional touchpad. Conventional computer pointing devices typically provide simultaneous control of only two continuous parameters and can process only a single point of contact. In addition, the way in which the rich touchpad is operated is much more natural and intuitive than the way in which conventional pointing devices are operated.

The touchpad can be used as a human-machine interface in a wide variety of applications, including CAD/CAE workstation control, real-time machine control, human-machine interfaces for the handicapped and electronic musical instruments. In addition, as mentioned earlier, the touchpad can also be used in machine sensing and robotics applications. The touchpad naturally lends itself to metaphors useful in a wide range of user interface applications and is simple to use, and its capabilities can easily be extended. Because of the touchpad's general-purpose nature and flexible reconfiguration capabilities, one basic system can be adapted for a wide range of applications.

This whitepaper is organized as follows. Section 2 provides illustrations of the user-level operation of the touchpad. Section 3 describes the system architecture and some basic implementations of the touchpad. Section 4 describes modular implementations of the touchpad, as well as some other alternatives to the basic implementations described in section 3.

The user-level operation of the touchpad and its many possible applications are touched on only briefly in this whitepaper. For detailed information, see the companion whitepaper [2], as well as U.S. Patents 6,570,078 [1] and 6,689,947 [4]. For detailed information about music and performance applications, see the companion whitepaper [3], the patents just men-

tioned, and the pending pre-grant patent publications U.S. 2004/0099131 [5], U.S. 2004/0074379 [6] and U.S. 2004/0069125 [7].

## 2 Illustrations of User-Level Operation

Before we consider the technology of the touchpad, it will be useful to give a sense of how it operates at the user level. We will first show how the touchpad makes it possible to control six parameters at once by inducing variations in a single contiguous image, and will then briefly discuss the touchpad's capability to process compound images, images consisting of multiple, non-overlapping contiguous images.

### Controlling Six Parameters with One Finger

Assume the sensor array is contacted by the end joint of a single finger, as suggested in Figure 1a. The pressure image produced by such contact will be similar to that shown in Figure 1b. Note that the darker the image, the higher the pressure.

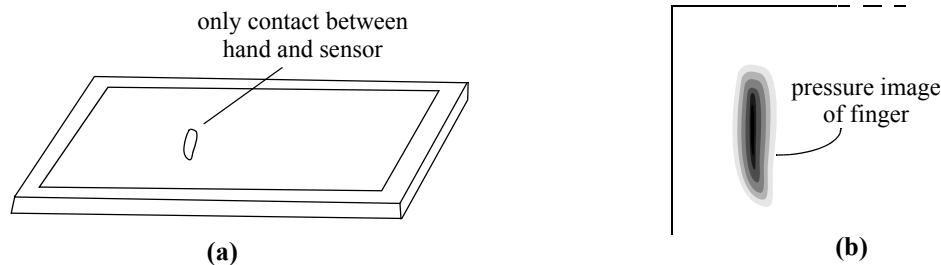
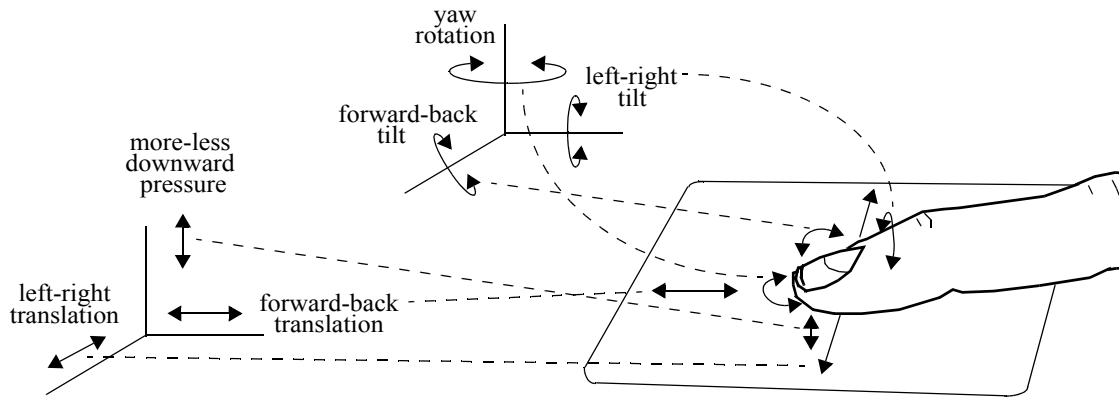


Figure 1. Contact of Finger with Pressure-Sensor Array

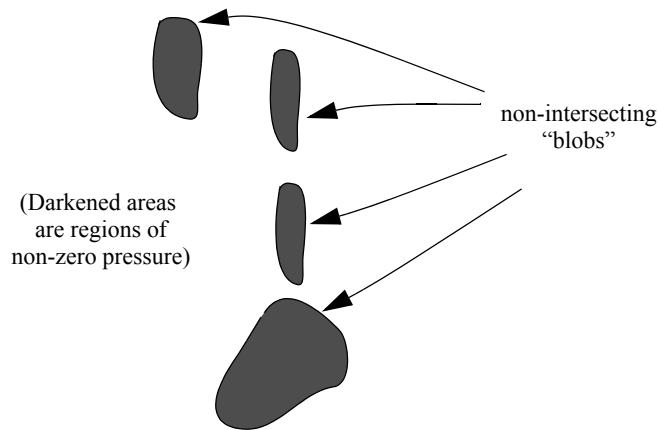
The finger can contact the sensor array anywhere on its surface, so the geometric center of the image determines two parameters -- the center's x and y coordinates -- that are easy to control independently of one another. The roughly elliptical form of the image has a measurable angular orientation, which the user can vary by moving her wrist so the finger pivots around its point of contact with the sensor array. It is also possible to control two dimensions of the finger's tilt, the degree to which pressure is concentrated with respect to the left-right axis and the front-back axis. Finally, the user can readily learn to keep all these parameters relatively constant while varying the average or total pressure exerted, giving a sixth independent parameter. Thus the touchpad enables a single finger to simultaneously control six independent parameters in a way that is very easy to learn. By varying the values of the parameters, the user can vary corresponding parameters of an arbitrary external device. Figure 2 illustrates the six degrees of freedom just described.

### Compound Images

A notable feature of the touchpad is its capability to process, and extract the values of parameters from, compound images. Figure 3 shows a compound image created by pressing the left part of the left hand against the sensor array with the palm raised. The image consists of four non-overlapping contiguous regions or "blobs."



**Figure 2. Six Parameters Can Be Controlled Simultaneously with One Finger**



**Figure 3. A Sample Compound Image**

In general, contact between the pressure-sensor array and multiple parts of the hand forfeits some degrees of freedom but introduces others. For example, if the end joints of two fingers are pressed against the sensor array as in Figure 3, it will be difficult or impossible to induce variations in the image of one of the end joints in six different dimensions (as described in the previous subsection) while keeping the image of the other end joints fixed. On the other hand, there are other parameters that can be varied, such as the distance between the two end joints. Compound images can provide control over many more parameters than a single contiguous image can. For example, by using the whole hand pressed flat against the sensor array, it is possible to vary as many as 17 parameters independently of one another.

This completes our brief discussion of the user-level operation of the touchpad. For more information, see [1,2].

## 3 System Architecture

We turn now to the architecture of the touchpad. We will begin by considering, in the first two sections, some aspects of the hardware design. We will next consider the overall information flow of the touchpad, and then some aspects of the software (or firmware) design. This section is concerned with the design of the basic system. In section 4, we will consider some enhancements of, and alternatives to, the basic system.

### Physical Formats

Figures 4a-d illustrate some possible physical formats for the touchpad. In Figures 4a-b the pressure-sensor array and its supporting hardware share the same housing. In Figure 4c the sensor array and at least some of its supporting hardware are separately housed and connected by a flexible cable to permit a smaller, more portable housing for the sensor array. In Figure 4d the supporting hardware is incorporated into a larger housing that also contains the system the touchpad controls (e.g. a computer workstation, a robotics system, an electronic musical instrument, a lighting control system). Many other formats are possible. For instance, a pressure-sensor array can be installed inside or on the surface of a glove. Similarly, a pressure-sensor array can be installed inside or on the bottom of a shoe or sandal, or inside or on the surface of a sock.

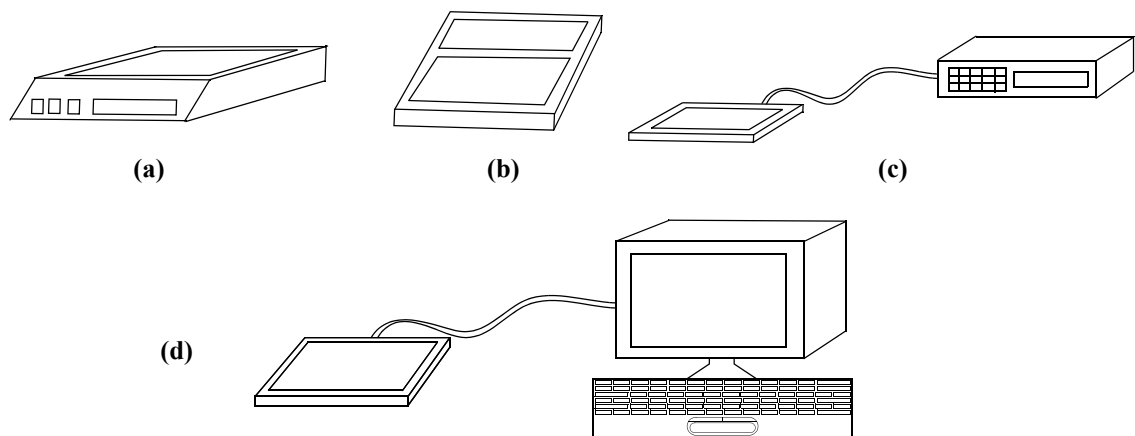
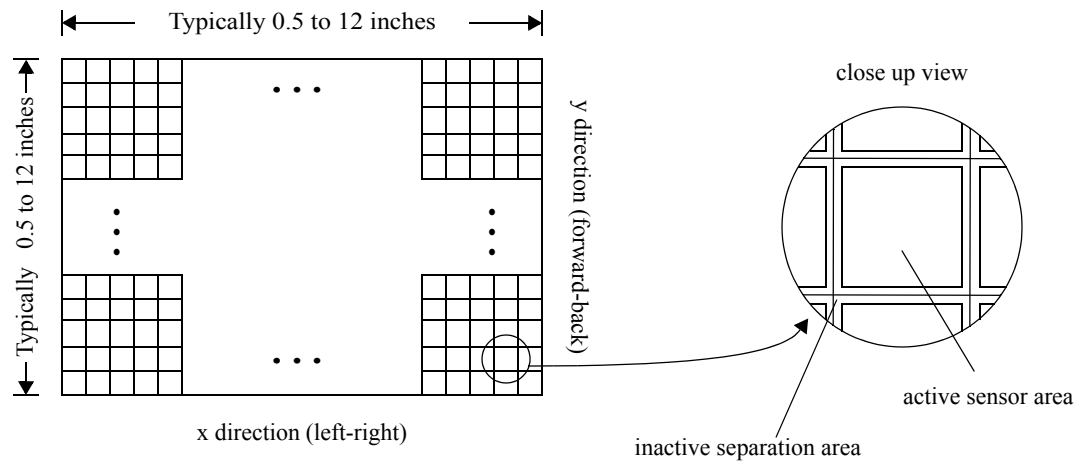


Figure 4. Examples of Physical Formats for the Touchpad

### Pressure-Sensor Array

A pressure-sensor array, depicted in Figure 5, is a central component of the touchpad. Although the design of the pressure-sensor array can be varied to accommodate functional limitations of the sensors and the requirements of particular applications, the following is a good working description given the current state of elastic-resistive compounds. A representative pressure-sensor array suitable for the touchpad has the following characteristics:

1. an area ranging from less than one square inch (for use on push buttons, PDAs, musical instrument keys, etc.) to over 100 square inches where large movements or high resolution is desired



**Figure 5. Pressure-Sensor Array**

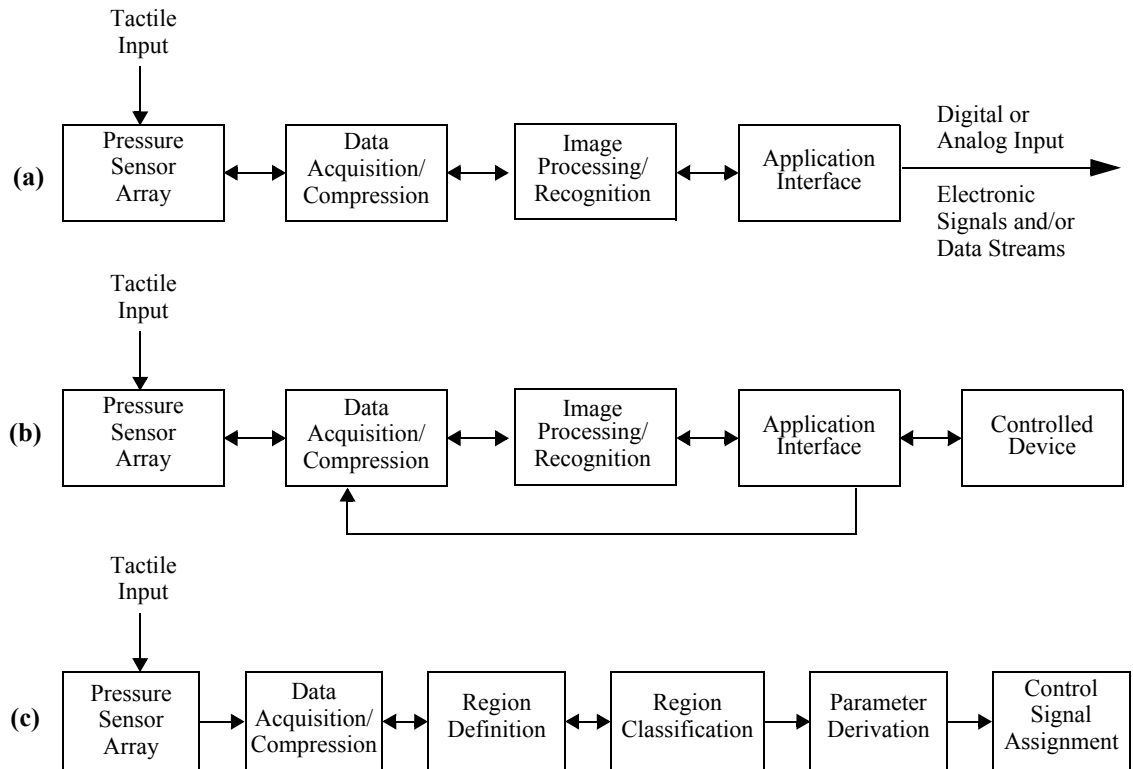
2. a spatial resolution for individual sensors of 1-2 square millimeters
3. a maximum of 25% of the length of the sides of the sensors as an inactive area between sensors
4. a durable, possible replaceable covering (such as a rubber mat) providing a comfortable contact surface
5. good mechanical integrity and reliability with a long operating life
6. quick elastic response time with freedom from mechanical resonances
7. resistive, capacitive, optical or other pressure-to-electronic measurement sensors capable of rapid scanning of pressure values
8. characteristics and interconnections permitting scans of the entire array at rates of 10-100 (nominally 20-40) times per second
9. 4-8 bits of non-hysteretic dynamic range in the electrical measurement (i.e. nominal hysteresis in this range), depending on the application requirements

Characteristics 1-4 are illustrated in Figure 5. Characteristics 7-9 are electrical characteristics critical to the rapid acquisition of accurate, usable information from the pressure-sensor array. Characteristic 8, in particular, influences the bussing, multiplexing and A/D conversions involved in the scanning. Note that it may be advantageous to use buffer amplifiers, pauses for settling times and concurrent scanning techniques, and that a number of different scanning techniques can be used. For more information, see the subsection “Segmented Busses and Pipelined-Interleaved Scanning” in section 4 below.

### Information Flow

Figures 6a-c show some possible high-level flows of information for the touchpad. They start with the reception of tactile information as input and end with the generation of control or data signals as output, or effects on an external device. Though information can flow in

only one direction from the reception of input to the production of output, it will typically be beneficial to use feedback, as shown in the figures.



**Figure 6. Sample Information Flows**

The information flow begins with the scan of the pressure-sensor array under the control of a data acquisition stage. Though the data acquisition stage need not incorporate a data compression capability, it can be advantageous to do so, as described below. The data acquisition stage looks for sensors whose pressure values exceed a low-level noise-rejection/deformity-rejection threshold. Pressure values that exceed the threshold are identified along with the location of the corresponding sensor, known via the sensor address. This information can be stored “raw” and processed later, or it can be subjected to simple adjacency tests and folded into running calculations, such as those described below.

The pressure values that exceed the threshold, and the locations of the sensors that have them, are then subjected to further processing:

1. Contiguous regions of the sensors are defined. A number of well-known simple, run-time adjacency tests can be used (see, for example, [8-11]).
2. The contiguous regions are subjected to classification tests and, if appropriate, are associated with regions defined in previous scans. In some cases, a single contiguous region may be divided into multiple intersecting but independent regions.
3. The values of various parameters are derived for each independent region. Examples of these parameters include the geometric center of the region, the

center of pressure, the average pressure, the area, the angle-of-rotation from reference for non-round regions, second- and higher-order pressure moments, the occurrence and features of specific types of events, and control parameters.

4. The values of relevant parameters are assigned to control signals, which are output to a signal routing, processing and synthesis entity. The control signals can be in a variety of formats.

Steps 1-3 fall in the category of image processing and can be implemented in a number of ways. One possibility is to do a considerable amount of the image processing in the data acquisition stage using techniques described below. Another possibility, illustrated in Figures 6a-b, is to do all the image processing in a single stage. Yet a third possibility, illustrated in figure 6c, is a more modular architecture in which the image processing functionality is implemented in three separate stages. Prior to their assignment to control signals, the derived values of the parameters can be remapped to create a more natural or intuitive response to the tactile input, as discussed below in the “Application Interface” subsection. For additional information about the information flow of the touchpad, see [1].

### **Data Acquisition and Data Processing**

The touchpad is intended as a real-time device with a fast response time. A problem that must be faced in implementing such a system -- i.e. one that must process a large number of measurements very quickly -- is memory access within the data processing hardware. By using special hardware and algorithms, the functions involved can be carried out in real time in an inexpensive implementation using existing, low-cost commodity microprocessors. The hardware and algorithms permit a large sensor array to be scanned and processed at rates of 10-100 (nominally 20-40) times per second. In this subsection we sketch such an algorithm, and in the next subsection we describe the algorithm in more detail.

The amount of information required to characterize each scan of the sensor array can be minimized by exploiting the expected characteristics of pressure images. A record comprising only a few entries is allocated for each “blob” (i.e. contiguous region of sensors with non-zero pressure values) when it is first detected. (Strictly, the relevant pressure values are ones that exceed a noise threshold, though we will say that such pressure values are non-zero.) As the scan progresses, the entries in the record undergo running updates. At the end of a scan, or upon confirmation that the record is complete for a given blob, simple post-scan processing is done on the record to calculate values of parameters useful for image processing and recognition. This technique considerably reduces the amount of information required to characterize each scan of the sensor array.

If an image contains  $N$  contiguous regions of non-zero pressure and the records have  $M$  entries, only  $N \times M$  fixed-point numbers must be processed when the scan is complete, regardless of the number of sensors in the array. For  $N=4$  and  $M=6$ , computations on only 24 numbers must be performed, compared to hundreds or thousands if the entire image is cached for analysis after the scan is complete. This approach can be described in more detail as follows:

- Each non-zero sensor measurement is assigned to exactly one new or existing record.

- The assignment depends on whether the sensor is adjacent to a sensor whose pressure value is already assigned to a record. If it is, its value is assigned to that record. Otherwise, a new record is created. Records are merged if later parts of a scan show that regions which seemed isolated in earlier parts of a scan are connected.
- When a non-zero sensor measurement is assigned to a record, the entries in the record are updated as functions of their prior value, the new sensor value and the coordinates of the sensor.
- At the end of a scan, the final values of the entries in each record are rapidly calculated.

Thus, as measurements are made during a scan, new records are created and existing records are incrementally updated, and final processing occurs when the scan is complete. Each record will typically contain between four and eight entries (depending on the precision the application requires), and all records for a given scan will have the same number of entries. On this approach, the results of a scan of 4096 sensors can be represented using a small number of total entries; for a typical number of records with a typical number of entries, the total number of entries across all records can be expected to be well under 20 in most cases.

A number of hardware approaches are possible, ranging from implementations using a shared microprocessor to implementations exploiting processing by special front-end hardware before information is presented to the microprocessor. When implemented using a pre-processing front stage, this 200:1 compression scheme saves tremendous amounts of memory, bandwidth and microprocessor cycles. For additional information, see [1].

### **Data Processing for a Single Region of Contact**

In this subsection, we describe the algorithm presented in the previous subsection in more detail. For simplicity, we assume a single region of contact is allowed. In the next subsection, we consider the case in which multiple regions of contact are allowed.

As each sensor is scanned, its pressure value is compared to a threshold that is used to interpret the measured value as either zero or non-zero. When the first sensor whose value is non-zero is encountered, a record associated with the current scan is created. Since only one contiguous region of contact is allowed, all non-zero measurements will be assigned to the same record, and no record assignment logic will be needed; sensors with non-zero pressure values that are not contiguous to other sensors with non-zero pressure values can be used to detect improper use of the touchpad.

In the case we are considering, the record for the region of contact contains six entries:

- N the number of sensors in the region
- X the sum of the x coordinates of the sensors in the region
- Y the sum of the y coordinates of the sensors in the region
- U the sum of the products of the x coordinate of each sensor in the region and its pressure value

- V the sum of the products of the y coordinate of each sensor in the region and its pressure value
- W the sum of the pressure values of all sensor in the region

(Note that, to determine whether a newly-scanned non-zero sensor is adjacent to some other non-zero sensor, the coordinates of some non-zero sensors must be included in the records. However, they need not be retained for the entire duration of the scan, but can be discarded as the scan progresses.)

The records used can be tailored to the particular type of contact. For instance, one type of record can be used for contact with a fingertip, another type for contact with a flat finger, and so on. Alternatively a generalized record format can be used that supports a large number of types of contact. Preliminary results suggest records containing as few as twelve entries can accommodate almost all practical possibilities.

When a scan is completed, the entries in the record are manipulated as follows to obtain the values of the listed parameters:

- $\frac{X}{N}$  the x coordinate of the geometric center of the region of contact
- $\frac{Y}{N}$  the y coordinate of the geometric center of the region
- $\frac{(WX - U)}{N}$  the x coordinate of the displacement of the “center of tilt” from the geometric center of the region
- $\frac{(WY - V)}{N}$  the y coordinate of the displacement of the “center of tilt” from the geometric center of the region
- $\frac{W}{N}$  the average pressure across the region

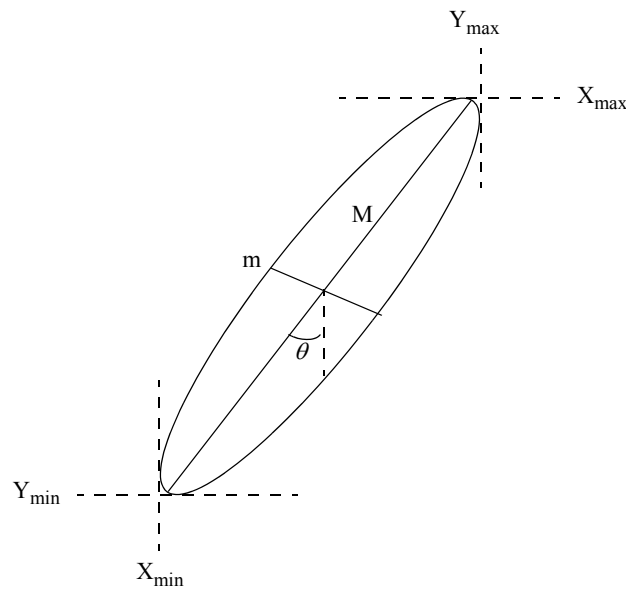
For this application, N, which represents the area of the region of contact, can be compared to a threshold value to identify and filter out contiguous regions of non-zero pressure that are too large or too small compared to the expected range of sizes of regions of contact.

There are a number of methods for calculating the orientation of the region of contact, all of which require additional entries in the record. Although more efficient and robust methods are preferable for actual implementations of the touchpad, the following will suffice for the purposes of exposition:

- $X_{\min}$  the minimum value of the x coordinates of all sensors in the region of contact
- $X_{\max}$  the maximum value of the x coordinates of all sensors in the region
- $Y_{\min}$  the minimum value of the y coordinates of all sensors in the region
- $Y_{\max}$  the maximum value of the y coordinates of all sensors in the region

As shown in Figure 7, the absolute value of the angle of orientation can be found by any of three different equations. (M and m are, respectively, the lengths of the major and minor axes of the elliptical region, obtained using a single training image.)

Since the shape of the region may vary slightly with rotation and tilting, m and M can be averaged, or other values can be extrapolated from them, for a more robust calculation of



$$\begin{bmatrix} (X_{\max} - X_{\min}) \\ (Y_{\max} - Y_{\min}) \end{bmatrix} = \begin{bmatrix} \sin \theta & \cos \theta \\ \cos \theta & \sin \theta \end{bmatrix} \begin{bmatrix} m \\ M \end{bmatrix}$$

$$\theta = \arctan \left[ \frac{(X_{\max} - X_{\min})m - (X_{\max} - X_{\min})M}{(Y_{\max} - Y_{\min})m - (Y_{\max} - Y_{\min})M} \right]$$

$$\theta = \arcsin \left[ \frac{(X_{\max} - X_{\min})m - (X_{\max} - X_{\min})M}{m^2 - M^2} \right]$$

$$\theta = \arccos \left[ \frac{(Y_{\max} - Y_{\min})m - (Y_{\max} - Y_{\min})M}{m^2 - M^2} \right]$$

**Figure 7. Calculation of Ellipse Orientation**

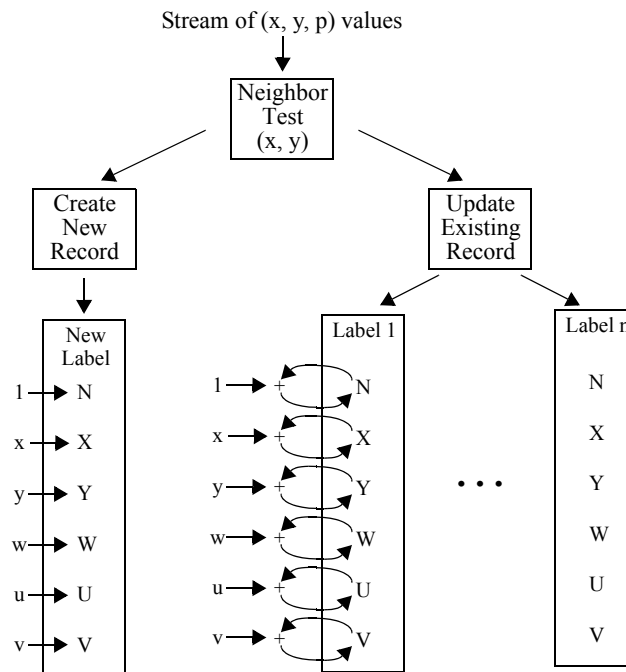
the angle of orientation. The sign of the angle of orientation can be stored in the record by using an additional binary entry that is set to one of two values according to the sign. A simple algorithm can be used to determine the sign: if the y coordinate of  $X_{\max}$  is greater than the y coordinate of  $X_{\min}$  then the angle is negative, and if the y coordinate of  $X_{\max}$  is less than the y coordinate of  $X_{\min}$  then the angle is positive.

The calculations just described can be refined. For example, if the orientation angle is calculated, tilt values can be adjusted as a function of the orientation angle to make it easier for the user to keep them constant while rotating the region of contact. Such calculations can be further refined by taking account of factors like the overall pattern of contact of the hand with the sensor array. (A spreadsheet demonstration model [12] of the derivation of the values of sample parameters from a pressure image, and explanatory documentation [13], are available from NRI<sup>®</sup> upon request.)

## Data Processing for Multiple Regions of Contact

In the last subsection we considered data acquisition and data records where only a single region of contact is allowed. In this subsection we extend the methods of the last subsection to accommodate multiple regions of contact.

The scan of the pressure-sensor array produces a stream of ordered triples of data, consisting of a pressure measurement for each sensor and its x and y coordinates. The pressure measurement is compared to a noise threshold and is classified as non-zero if it exceeds the threshold and zero otherwise. Each time a sensor is encountered with a non-zero pressure, its x and y coordinates are subjected to an adjacency test to determine whether the sensor is a neighbor of a previously-scanned sensor with a non-zero pressure. If it is not, a new record is created. Otherwise, a record assignment logic assigns the sensor to the appropriate record, identified by a label, and the record is updated on the basis of the ordered triple associated with the sensor.



**Figure 8. Processing of Data Stream from Scan for Multiple Regions of Contact**

As illustrated in Figure 8, in the case we are considering an existing record is updated as follows:  $N$ , the area of the region of non-zero pressure, is incremented by 1; the x coordinate of the just-scanned sensor is added to  $X$ , the sum of all the x coordinates of the sensors in the region; the y coordinate is added to  $Y$ , the sum of all the y coordinates of the sensors in the region; the pressure is added to  $W$ , the sum of all the pressures of the sensors in the region; the product of the x coordinate and the pressure is added to  $U$ , the sum of the products of the x coordinate of each sensor in the region and its pressure; and the product of the y coordinate and the pressure is added to  $V$ , the sum of the products of the y coordinate of each sensor in the region and its pressure.

The scan may reveal that two or more regions of non-zero pressure that appeared to be separate are in fact foliations of a single, larger, contiguous region. If so, the records for the region will be merged or marked to be merged later. In the case we are considering, since each entry in the record is a running sum, merging records is simply a matter of adding the values for the corresponding entries in each record. At the end of the scan, each record is processed separately to compute the value of parameters useful for controlling an external system. Those values are assigned to control signals, which are then transmitted to the external system.

### Image Recognition

It may be useful to use different kinds of contact with the pressure-sensor array to carry out different functions. For instance, in a CAD/CAE application, a fingertip might be used to select an icon from a palette, and a finger end joint to position the icon in the drawing area. (For more detailed examples, see [2].) In such cases, it will be necessary to distinguish the different kinds of contact produced by different parts of the hand. We describe how this can be done in this subsection.

In the example we now consider, the user can provide input to the touchpad using any of the following parts of her hand:

- a fingertip
- a finger end joint
- a thumb end joint
- a wrist

Images created using each of these parts of the hand are shown in Figure 9. For simplicity, we assume the user controls only three parameters with each region of contact:

- the x coordinate of the geometric center of the region
- the y coordinate of the geometric center of the region
- the average pressure of the region

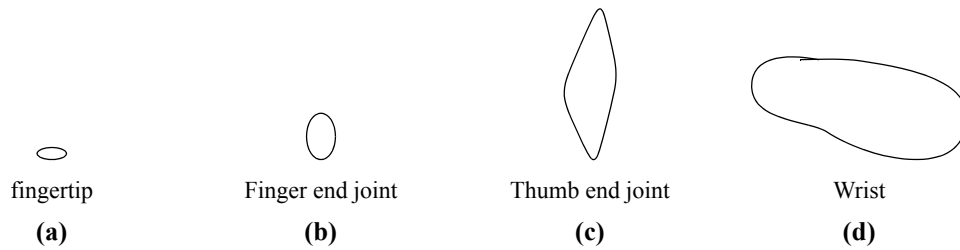
Each region of contact is represented with its own record. The records might consist of the following entries:

- N the number of sensors in the region
- X the sum of the x coordinates of the sensors in the region
- Y the sum of the y coordinates of the sensors in the region

The records are updated as the scan of the sensor array progresses, as described in the previous subsection. As Figure 9 shows, the areas of the regions created by contact with each of the parts of the hand in the above list stand in the following relationship:

Fingertip < Finger end joint < Thumb end joint < Wrist

This relationship makes it possible to distinguish the types of contact we are considering using a very simple procedure: by considering the value of N. To recognize other types of contact, such as contact with a palm or fist, more complicated procedures may be needed.



**Figure 9. Outlines of Regions of Contact Created by Different Parts of the Hand**

However, in many cases, very robust image recognition can be achieved by adding to the record for each region of contact the entries  $X_{\min}$ ,  $X_{\max}$ ,  $Y_{\min}$  and  $Y_{\max}$  (defined above) and submitting them to a few simple tests at the end of each scan.

### Application Interface

The application interface maps the raw parameters extracted from pressure images to an output format, such as MIDI messages, suitable for controlling an external device. The particular format that is used can be programmable, and multiple format specifications can be stored so the user can select a particular format for a particular application. The raw parameters can be scaled or warped to create a more natural or intuitive response to the tactile input. They can also be quantized into two or more discrete steps that can, for instance, be interpreted as sequential notes of a musical scale or melody.

The ways in which the raw parameters are scaled or warped can be fixed for all applications. Alternatively, they can be application-specific and retrieved from memory each time a particular application is used. Yet another possibility is for them to vary in real time as a function of features of particular kinds of pressure images. For instance, parameters can be scaled in one way for images that include only a finger end joint, in another way for images that include only a thumb end joint, and in a third way for images that include both a finger end joint and a thumb end joint.

Other functions the application interface can perform include:

- assignment of parameters to output channels
- temporal smoothing (i.e. low-pass filtering) and holding
- threshold detection and logic operations for logic-oriented outputs
- channel assignments
- serialization, packetization or multiplexing of output data
- packet buffering and protocol
- D/A conversions

### Immunity to Sensor Failures

The touchpad derives its output from regions of contact containing multiple pressure sensors. For a sensor array whose resolution falls within the typical range, even the smallest

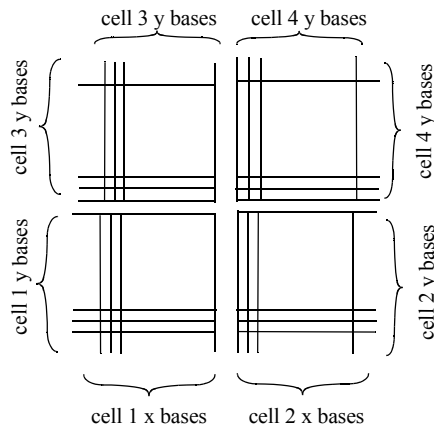
region of contact will contain on the order of ten sensors. As a result, redundancy in the pressure images created by contact with the sensor array can be exploited to make the touchpad's operation immune to isolated sensor failures -- for instance, by averaging or smoothing the pressure measurements of multiple neighboring sensors. Alternatively, an isolated sensor failure can be identified by out of trend values over time and filtered out during data acquisition. Further, since the technique for run-time compression uses running sums, the measurements of one or more errant sensors can be omitted with minimal error.

## 4 Modularity and Alternative Implementations

In the last section we considered the architecture and design of the basic touchpad system. In this section we consider some enhancements of and alternatives to the basic system.

### Segmented Busses and Pipelined-Interleaved Scanning

In this subsection we consider how pressure-sensor arrays can be partitioned to facilitate data acquisition. In the next subsection we consider a related case, the ganging of sensor arrays to create a larger array.

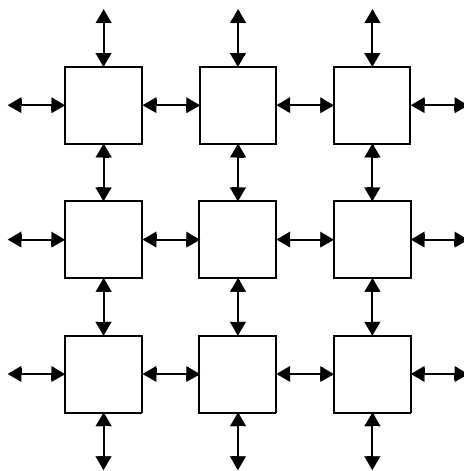


**Figure 10. Pipelined-Interleaved Scanning Arrangement for Pressure-Sensor Array Busses**

A pressure-sensor array can be divided into partitions by splitting electrical scanning interconnection busses into separate segments, as shown in Figure 10. Splitting the busses in this way makes it possible to use a number of “concurrent” or “pipelined-interleaved” scanning techniques, and reduces the computational and electronics requirements for rapid scanning of the sensor array; for instance, settling time requirements are eased and bus electrical capacitance is reduced. For additional information, see [1].

The streams of data produced as the array is scanned can be processed in parallel or merged. The latter can be done in such a way that the same data processing techniques described in section 3 above can be used. In either case, the partitions of the sensor array need to share information. Figure 11 illustrates a signal flow that accomplishes this. In this case, each

partition is electrically (or optically) connected to its immediate neighbors, though partitions can be interconnected in other ways, such as via busses or star-topologies.



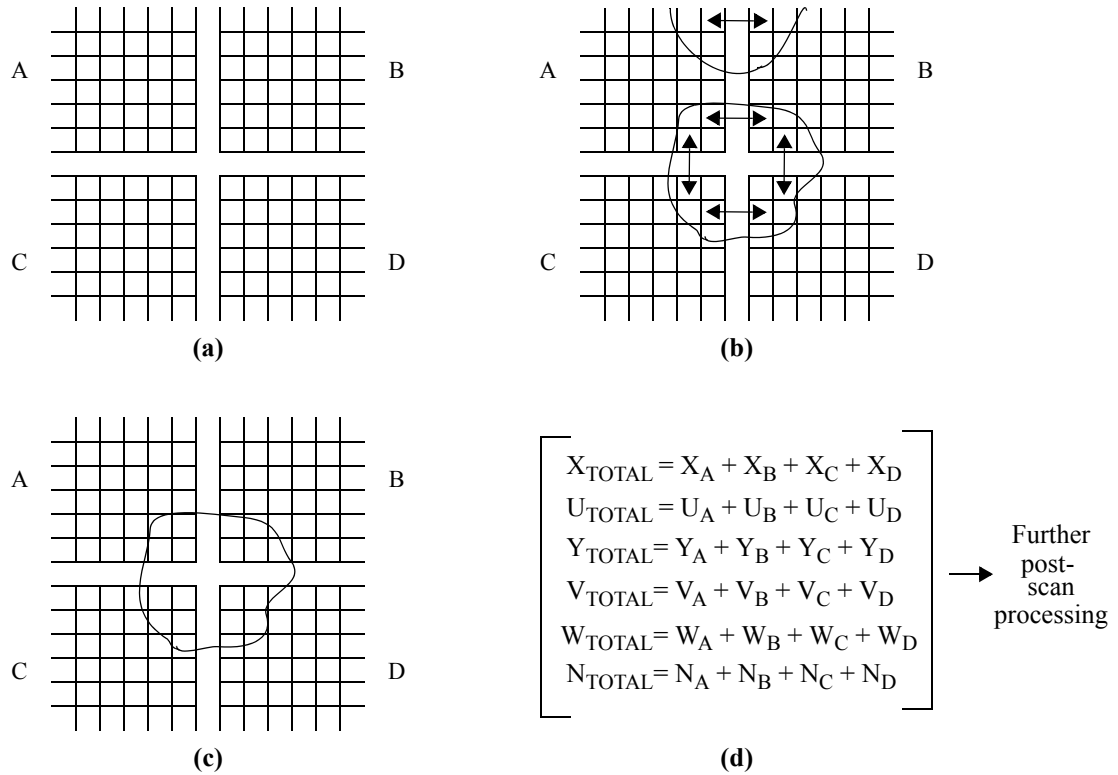
**Figure 11. Signal Flows for Exchanging Information Between Partitions**

A problem that arises for partitioned sensor arrays is how to handle contiguous regions of contact that span two or more partitions. Consider an array with partitions A, B, C and D, as shown in Figure 12a. In Figure 12b two contiguous regions of contact that span multiple partitions are depicted, with the top region spanning partitions A and B, and the bottom region spanning partitions A, B, C and D. In each case, the partitions the regions span share and pool information via the flows shown in Figure 11, as suggested by the arrows in Figure 12b.

To see how this can be done, consider the case of a contiguous region that spans all four partitions, as shown in Figure 12c. On one approach, each partition creates its own record for the portion of the contiguous region it detects and updates the record as the scan progresses. The records are merged after the scan is complete by adding the corresponding running sums in all the records. On another approach, a single record is created for all four partitions, and, as the scan progresses, the relevant values for each partition are added to the appropriate running sums in the record. On both approaches, the ultimate result is a single record with entries that are sums of the relevant values for each partition, as shown in Figure 12d.

### **Ganging Sensor Arrays**

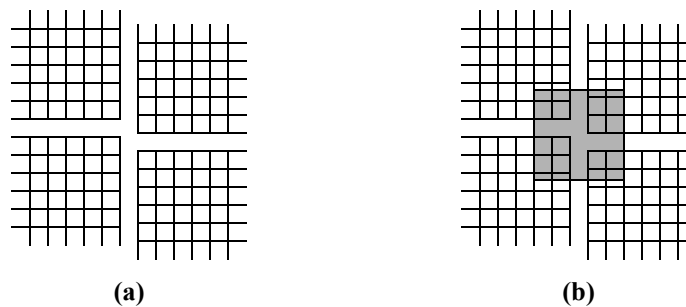
In implementations of the touchpad in which the pressure sensors are sequentially scanned, it will often be advantageous to perform the scan periodically or algorithmically, as is done with light sensing elements in a CCD camera. For large sensor arrays, this would require the additional cost and complexity of high-performance electronics, EMI shielding and high computational speeds. In implementations of the touchpad in which the sensors are scanned in parallel, scaling to large sensor arrays would also introduce additional cost and complexity. In addition, the fabrication of large sensor arrays would be costly and likely to suffer from yield problems, and damage in the field would also be costly. A further problem arises from the requirement of some applications to have sensor arrays of unusual shapes



**Figure 12. Handling Contiguous Regions that Span Multiple Partitions**

or aspect ratios. This could be the case, for instance, for sensor arrays affixed to musical instrument keys and musical instrument bodies. For these reasons, methods for ganging multiple small pressure-sensor arrays to behave collectively as one large one are of interest.

The same methods described in the previous subsection for sharing information among partitions, and for handling contiguous regions of contact that span multiple partitions, can be used to gang sensor arrays. There is, however, an additional problem that must be faced: how to handle cases in which individual sensor array modules are misaligned. An example, in which four sensor array modules are misaligned, is shown in Figure 13a. If the degree of misalignment is small enough relative to the sensor size, only a negligible error will result. But in cases where the degree of misalignment is larger, additional capabilities will be needed to correct for it.



**Figure 13. Misaligned Pressure-Sensor Array Modules**

One way to do this is to calibrate the composite sensor array using an object of known size and shape, as illustrated in Figure 13b. The object is placed on the composite sensor array so that it covers the area where the modules meet. Each module determines which sensors are under the edges of the object, and the modules communicate with one another to associate the sensors under one edge of the calibration object in one module with the sensors under the same edge in the adjacent module. Once these associations have been made, the coordinates of the sensors in each module can be adjusted to correct for the misalignment. In some cases, the misalignment may result in a gap between modules. The calibration procedure can determine the size of the gap, and, if it is not too large (no more than a few sensors in size), it can be treated in the same fashion as the failure of isolated sensors described in the last section.

### User-Level Partitioning

We have seen how, through splitting scanning interconnection busses into separate segments, the computational and electronic requirements for rapid scanning of the sensor array can be reduced. It is also possible to partition the sensor array at the user level; that is, the surface of the array is divided into multiple, functionally-distinct regions, as illustrated in Figure 14. Each partition operates independently of the others, and different partitions can respond to the same kind of contact in different ways. The software/firmware techniques supporting user-level partitioning are closely related to those supporting pipelined-interleaved scanning and the ganging of sensor arrays. For more information about user-level partitioning, see [2,3].

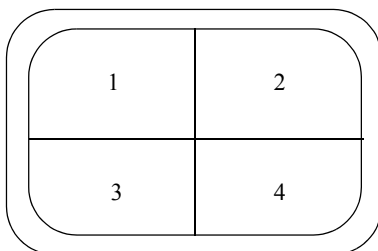


Figure 14. User-Level Partitioning of a Pressure-Sensor Array

### Visual Displays

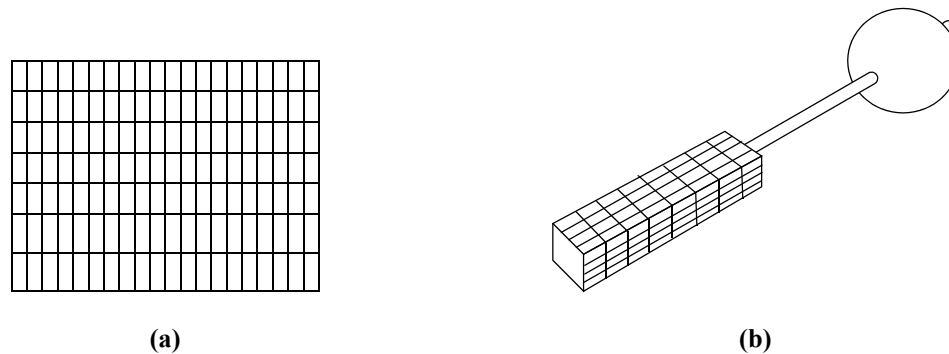
If the pressure sensors in a pressure-sensor array can be fabricated to be transparent or translucent, an LCD, LED or other display can be positioned under the array. Alternatively, each sensor in the array can have an integrated LCD or LED under its pressure-sensing component, or LCDs or LEDs can be interspersed among the sensors in a regular lattice.

One important use for a visual display is to indicate the borders of partitions when user-level partitioning is used. The display can be used to indicate the borders with lines, or each partition can have a different color. The partitions can also be labeled with text.

### Chip Implementations

It is possible to package a small pressure-sensor array with data acquisition hardware and a small processor to create a low-profile chip. The chips can be laid as tiles in an array, as

shown in Figures 15a-b. Advanced seating and connector technologies, as used in laptops and other high-performance miniature consumer electronic devices, can be used to minimize the separation between adjacent chips and to make the top surface of the tiled array smooth. Such chips can be crafted with the profiles of contemporary integrated circuit packages with bottom-interconnect. Creating sensor arrays by tiling chips will make maintaining the touchpad easier and lengthen its life.



**Figure 15. Arrays of Tiled Touchpad Chips**

Data processing for an array of chips can be performed by partitioning the array and using relatively simple distributed message-passing algorithms. The chips can direct their output streams to a processor that processes the data and assigns the parameter values that are derived to control signals. Alternatively, in place of a dedicated processor, there can be multiple types of chips. For instance, different types can be used to process the edges and the interiors of regions of contact, or one type can be general purpose while another type is dedicated to input/output. It is also possible for the functions of each type of chip to be programmable. This would enable the user to select the function for each type of chip that is best suited for its location in the chip array. In a more sophisticated implementation, the chips communicate among themselves to determine automatically how they are interconnected and what function is best suited to their location in the array.

An array of chips can be networked by a tapped passive bus; a one- or two-dimensional mode active bus daisy-chain; a centralized, expandable star-wired message-passing chip or other subsystem; or other means. The chips can be designed so that the physical connections needed for signal transmission are made when they are positioned next to one another. A flexible, durable film can be affixed to the surface of each chip to protect it from the outside world. Monochrome or color displays can be built into the chips under a transparent or translucent sensor, as described in the previous subsection.

The chips can be put on control surfaces, musical instrument keys or instrument bodies, mallet handles (as shown in Figure 15b), gloves, clothing, robot hands, tablets, work surfaces and so on. Because the chips can be used to create a wide variety of shapes, it is anticipated that they would have a large market. If so, there would be a significant economy-of-scale in their manufacturing, which would lower their cost and make extensive feature development feasible.

## Fabric Implementations

Through the use of “flexible electronics” fabrication techniques, a pressure-sensor array together with its supporting processor and electronics can be implemented as a flexible fabric. The processors, electronics, data processing software and wiring can be designed so that cutting the fabric at modular boundaries will allow unimpeded operation of the resulting pieces of fabric, and joining pieces of fabric will create larger, functional pieces. The fabric can further incorporate a visual display, as described above.

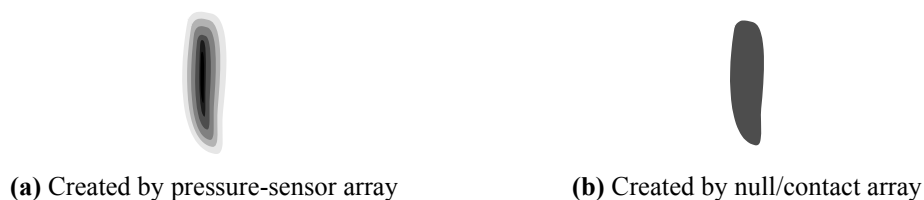
## Alternatives to Pressure Sensors

In this whitepaper, we have considered only contact surfaces incorporating sensors that measure pressure. But the electronics and data processing software can be adapted for use with arrays of other types of sensors, such as image, video, chemical and optical sensors. For more information, see [1,7,14].

## A Low-Cost, Limited-Capability Touchpad

In the early stages of productization, the pressure-sensor arrays and other components of the touchpad may be costly. Therefore, to create market acceptance and market demand for the touchpad, and to create familiarity with the basic ideas underlying its operation, it may be desirable to introduce a low-cost, limited-capability alternative while developing the full-featured version. One such alternative combines a null/contact touchpad with a subset of the data processing capability of the full-featured version [1,6]. Null/contact touchpads are used in laptop computers and touchscreens. Rather than having sensors that measure multiple gradations in pressure, null/contact touchpads have sensors that have only two states, “on” when the pressure exceeds a threshold and “off” otherwise.

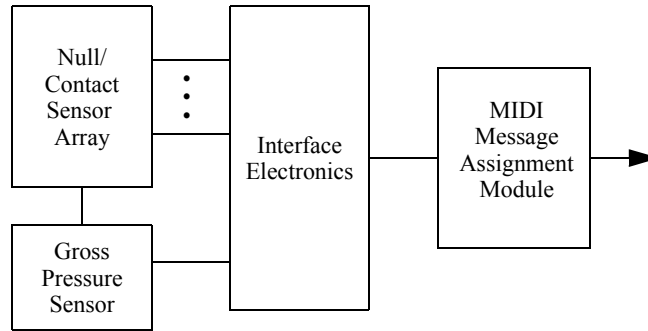
Figures 16a-b illustrate the difference between a null/contact sensor array and a pressure-sensor array. Both figures show an image created by the contact of the end joint of a finger with the sensor array. But while the pressure sensors in the full-featured version pixellate the image into an array of different pressure values, as illustrated in Figure 16a, the contact sensors in the null/contact touchpad pixellate the image into an array of on and off values, as illustrated in Figure 16b.



**Figure 16. Comparison of Images Created by Pressure-Sensor and Null/Contact Arrays**

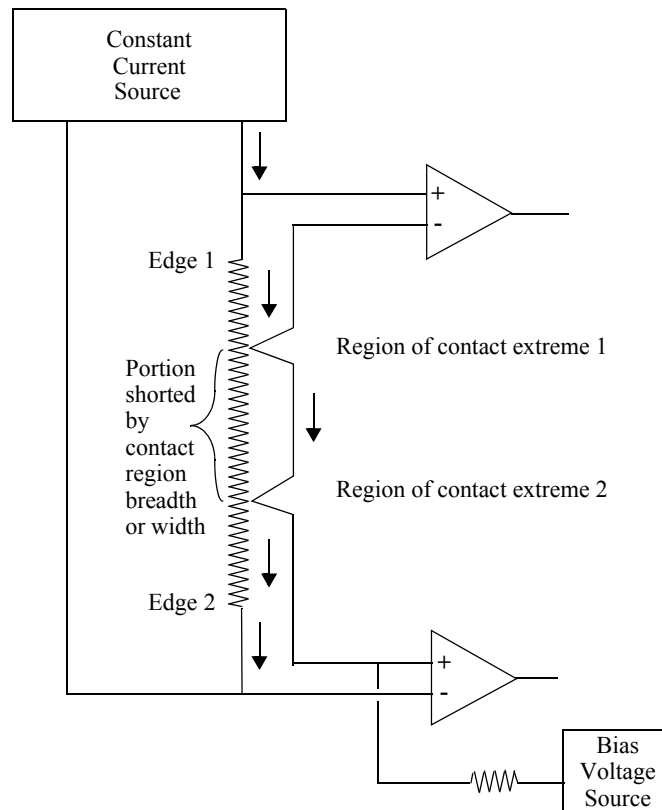
In its most common implementation, a null/contact touchpad comprises a layer of two resistive films, each of which makes contact with a metal foil whenever the pressure exceeds a certain threshold. Other implementations are possible: the null/contact touchpad can use capacitive detection or an array of on/off pressure sensors. Figure 17 shows an architecture that enables a null/contact touchpad to control MIDI devices. The interface electronics and

MIDI message assignment module can be adapted so that multiple null/contact touchpads, each fitted with its own gross pressure sensor, can share them.



**Figure 17. Null/Contact Touchpad Equipped to Send MIDI Messages**

A null/contact touchpad can be used to control three independent parameters: the left-right location of a region of contact, the forward-back location and the area. In addition, in some cases the user can control the shape, width and length of the region of contact. In the case of null/contact touchpads that employ resistive films or capacitive detection, there are highly efficient methods requiring minimal data processing that can be used to derive values for these additional parameters.



**Figure 18. Circuit for Detecting the Distance Across a Region of Contact**

Figure 18 depicts a circuit for one of these methods. In this example, a null/contact touchpad employing the resistive-film technology measures the maximum width (or maximum length) of a region of contact. To do so, two opposite edges of the resistive film are connected across a common current source. As a result, if the region of contact has a non-negligible width (or length), a portion of the resistive film is shorted by the metal foil. By Ohm's law, the voltage between each of the edges of the resistive film and the metal foil is proportional to the distance between the edge and the closest point of the region of contact. To measure both the maximum width and the maximum length of a region of contact, two instances of the circuit just described are used, one going left-to-right and the other front-to-back. Additional measurements, such as the edge-to-edge voltage across the entire film, can be used to calculate the values of other parameters, such as (for certain shapes of regions of contact) the area and the approximate geometric center.

## 5 Further Information

The purpose of this document has been to provide technical information about the NRI<sup>®</sup> rich touchpad. This document is based on U.S. Patent 6,570,078 [1] and related issued and pending patents [4-7,14], all licensable from NRI<sup>®</sup>. Detailed hardware and software reference designs can be discussed under negotiable terms. All financial or in-kind proceeds from such arrangements will be used to fund pure academic research at NRI<sup>®</sup>. Contact [inquiries@newrenaissanceinstitute.com](mailto:inquiries@newrenaissanceinstitute.com) for more information.

## References

- [1] NRI<sup>®</sup>, U.S. Patent 6,570,078, "Tactile, Visual, and Array Controllers for Real-Time Control of Music Signal Processing, Mixing, Video, and Lighting," May 27, 2003 (PDF available at [www.newrenaissanceinstitute.com/Issued.Patents-PDF/Tactile\\_Visual\\_and\\_Array.pdf](http://www.newrenaissanceinstitute.com/Issued.Patents-PDF/Tactile_Visual_and_Array.pdf)).
- [2] NRI<sup>®</sup> Whitepaper, "The NRI<sup>®</sup> Rich Touchpad: Operation and Applications," 2004 (PDF available at [www.newrenaissanceinstitute.com/WP\\_list.htm](http://www.newrenaissanceinstitute.com/WP_list.htm)).
- [3] NRI<sup>®</sup> Whitepaper, "The NRI<sup>®</sup> Rich Touchpad: Music and Performance Applications," 2004 (PDF available at [www.newrenaissanceinstitute.com/WP\\_list.htm](http://www.newrenaissanceinstitute.com/WP_list.htm)).
- [4] NRI<sup>®</sup>, U.S. Patent 6,689,947, "Real-Time Floor Controller for Control of Music, Signal Processing, Mixing, Video, Lighting, and Other Systems," February 10, 2004 (PDF available at [www.newrenaissanceinstitute.com/PreGrant-PDF/FloorController.pdf](http://www.newrenaissanceinstitute.com/PreGrant-PDF/FloorController.pdf)).
- [5] NRI<sup>®</sup>, U.S. Pre-Grant Patent Publication 2004/0099131, "Transcending Extensions of Classical South Asian Musical Instruments," May 27, 2004. (PDF available at [www.newrenaissanceinstitute.com/PreGrant-PDF/SouthAsian.pdf](http://www.newrenaissanceinstitute.com/PreGrant-PDF/SouthAsian.pdf)).
- [6] NRI<sup>®</sup>, U.S. Pre-Grant Patent Publication 2004/0074379, "Functional Extensions of Traditional Music Keyboards," April 22, 2004. (PDF available at [www.newrenaissanceinstitute.com/PreGrant-PDF/Keyboards.pdf](http://www.newrenaissanceinstitute.com/PreGrant-PDF/Keyboards.pdf)).

- [7] NRI<sup>®</sup>, U.S. Pre-Grant Patent Publication 2004/0069125, "Performance Environments Supporting Interactions Among Performers and Self-Organizing Processes," April 15, 2004 (PDF available at [www.newrenaissanceinstitute.com/PreGrant-PDF/SelfOrganized.pdf](http://www.newrenaissanceinstitute.com/PreGrant-PDF/SelfOrganized.pdf)).
- [8] Hara, Yoshiko, "Matsushita demos multilayer MPEG-4 compression," *Electronic Engineering Times*, April 19, 1999.
- [9] Viberg, Mats, *Subspace Fitting Concepts in Sensor Array Processing*, Linkoping Studies in Science and Technology. Dissertations No. 27, Linkoping, Sweden 1989.
- [10] Shapiro, Larry S., *Affine Analysis of Image Sequences*, Cambridge University Press, 1995.
- [11] Ronse, Christian and Devijver, Pierre A., *Connected Components in Binary Images: the Detection Problem*, John Wiley & Sons Inc., New York, 1984.
- [12] NRI<sup>®</sup>, "Touchpad Spreadsheet Model Software" (available by special arrangement; contact NRI<sup>®</sup> for information).
- [13] NRI<sup>®</sup>, "Touchpad Spreadsheet Model User's Guide" (available by special arrangement; contact NRI<sup>®</sup> for information).
- [14] NRI<sup>®</sup>, U.S. Pre-Grant Patent Publication 2004/0069125, "Controlling and Enhancing Electronic Musical Instruments with Video," April 15, 2004 (PDF available at [www.newrenaissanceinstitute.com/PreGrant-PDF/Video-Instrument.pdf](http://www.newrenaissanceinstitute.com/PreGrant-PDF/Video-Instrument.pdf)).