[0053] and m_{0a}=slope of the first edge seen by $Camera_0$
[0054] and m_{0b}=slope of the second edge seen by $Camera_0$
[0055] and m_{0c}=average of m_{0a} and m_{0b}
[0056] and m_{1a}=slope of the first edge seen by $Camera_1$
[0057] and m_{1b}=slope of the second edge seen by $Camera_1$
[0058] and m_{1c}=the average of m_{1a} and m_{1b}
[0059] and (x_{0a}, y_{0a})=the intersection of m_{0a} and m_{1c}
[0060] and (x_{0b}, y_{0b})=the intersection of m_{0b} and m_{1c}
[0061] and (x_{0c}, y_{0c})=the intersection of m_{0c} and m_{1c}, the touch center
[0062] and (x_{1a}, y_{1a})=the intersection of m_{1a} and m_{0c}
[0063] and (x_{1b}, y_{1b})=the intersection of m_{1b} and m_{0c}
[0064] as this is the same as (x_{0c}, y_{0c}) and r_0=the distance of the touch center from $Camera_0$
[0065] and r_1=the distance of the touch center from $Camera_1$
[0066] and w_0=the width or distance of point (x_{0a}, y_{0a}) to (x_{0b}, y_{0b})
[0067] and w_1=the width or distance of point (x_{1a}, y_{1a}) to (x_{1b}, y_{1b})
[0068] then to calculate the width (w_0) of the touch area as observed by $Camera_0$, the following equations are used:

$$x_{0a}=m_{1c}/(m_{0a}-m_{1c})$$

$$y_{0a}=m_{0a}*x_{0a}$$

$$x_{0b}=m_{1c}/(m_{0b}-m_{1c})$$

$$y_{0b}=m_{0b}*x_{0b}$$

$$x_{0c}=m_{1c}/(m_{0b}-m_{1c})$$

$$y_{0c}=m_{0b}*x_{0c}$$

$$r_0=sqrt(x_{0c}^2+y_{0c}^2)$$

[0069] Similar equations can be used to calculate the width (w_1) of the touch area as observed by $Camera_1$. After solving for width, the touch area (S) can be calculated using the following equation:

$$S=w_0*w_1,$$

[0070] where w_0 is the width of the touch area as detected from $Camera_0$ and w_1 is the width of the touch area as detected from $Camera_1$.

[0071] FIG. **6**, comprising FIG. **6**A and FIG. **6**B, shows a simple stylus **602** that has been modified to enable multiple touch areas based on applied pressure. The stylus **602** includes a spring loaded plunger **604** that is designed to collapse into the tip **606** of the stylus **602** when sufficient compression is applied to the spring **608**. Thus, when the stylus **602** is made to hover in proximity to the touch screen **110** or to contact the touch screen **110** without sufficient pressure to compress the spring **608**, the plunger **604** will remain protruded from the tip **606**. The detectors **130**, **131** will detect the presence of the plunger **604** and the computing device **201** will base the computation of touch area (S) on the detected size of the plunger. Conversely, when the stylus **602** is made to contact the touch screen **100** with sufficient pressure to compress the spring **608**, the plunger **604** will collapse into the tip **606**, which will itself contact the touch screen **110**. The computing device **201** will thus base the computation of the enlarged touch area (S') on the detected size of the stylus tip **606**.

[0072] The stylus **602** of FIG. **6** is designed to operate in a manner similar to a finger **302**, which creates an enlarged touch area when pressure is applied. Other stylus designs can accomplish similar functionality. For example, similar functionality could be provided by a stylus having a rubber tip that expands (area-wise) when pressure is applied to it. Accordingly, any stylus or other object that can be used to indicate both a smaller and a larger area can be used in accordance with embodiments of the present invention.

[0073] FIG. **7** is a flow chart illustrating an exemplary method **700** for discerning between a tracking state, a selection state and an out-of-range state. The method **700** begins at starting block **701** and proceeds to step **702**, where a determination is made as to whether a finger or stylus is detected in the energized plane proximate to the touch screen. If no finger or stylus is detected, the method advances to step **704**, where the interaction state is indicated to be "out-of-range". Following step **704** the method loops back to step **702** for further process. When a finger or stylus is detected at step **702**, the method proceeds to step **706**, where an image captured by a first detector is processed to determine approximate coordinates for a first pair of outer edges of the finger or stylus. For example, such coordinates may be determined using slope line calculations. Next at step **708**, an image captured by a second detector is processed to determine approximate coordinates for a second pair of outer edges of the finger or stylus. At step **710**, the approximated coordinates of the two pairs of outer edges of the finger or stylus are used to calculate an approximated touch area.

[0074] After calculating an approximated touch area at step **710**, the method proceeds to step **712** for a determination as to whether the approximated touch area is greater than a threshold touch area. The threshold touch area may be established through calibration of the touch screen system **100** or may be specified by a system operator or administrator. If the approximated touch area is greater than the threshold touch area, a selection state is indicated at step **712**. If the approximated touch area is not greater than the threshold touch area, a tracking state is indicated at step **714**. From either step **712** or step **714**, the method returns to step **702** for further processing.

[0075] As will be apparent to those of ordinary skill in the art, touch position calculations can be performed in sequence or in parallel with the calculations to approximate interaction state. Thus, if movement of the finger or stylus is detected while iterations through the exemplary method **700** indicate a continued selection state, the continued selection state will be recognized as a dragging state. Indication of a continued tracking state in conjunction with movement of the finger or stylus may be recognized, for example, as requiring a cursor to follow the finger or stylus.

[0076] FIG. **8** is a state diagram showing the operation sequence of certain exemplary embodiment of the present invention. The tracking state **802** is indicated when the user's finger or stylus is detected within the energized plane in proximity to the touch screen **110** and a calculated touch area is determined to be less than or equal to a threshold touch area. If the finger or stylus is not moving (i.e., detected velocity is approximately zero), the stationary state **804** is indicated. During the stationary state **804**, a threshold touch area can optionally be calibrated, for example as a background process. From the stationary state **804**, if the finger or stylus starts to move (i.e., detected velocity is greater than zero) and the calculated touch area remains less than or equal to the threshold touch area, the tracking state **802** is again indicated.