Swiss-Cheese Extended: An Object Recognition Method for Ubiquitous Interfaces based on Capacitive Proximity Sensing

Tobias Grosse-Puppendahl, Andreas Braun, Felix Kamieth, and Arjan Kuijper
Fraunhofer IGD, Fraunhoferstr. 5, 64283 Darmstadt, Germany
{tobias.grosse-puppendahl,andreas.braun,felix.kamieth,arjan.kuijper}@igd.fraunhofer.de

ABSTRACT
Swiss-Cheese Extended proposes a novel real-time method for recognizing objects with capacitive proximity sensors. Applying this technique to ubiquitous user interfaces, it is possible to detect the 3D-position of multiple human hands in different configurations above a surface that is equipped with a small number of sensors. The retrieved object configurations can significantly improve a user's interaction experience or an application's execution context, for example by detecting multi-hand zoom and rotation gestures or recognizing a grasping hand. We emphasize the broad applicability of the proposed method with a study of a multi-hand gesture recognition device.

Author Keywords
capacitive proximity sensing; capacitive sensing; 3D interaction; ubiquitous interfaces; object recognition; object tracking

ACM Classification Keywords
H.5.2. Information Interfaces and Presentation: User Interfaces - Graphical user interfaces; Input devices & strategies

General Terms
Algorithms; Measurement

INTRODUCTION

Twenty years ago, Ubiquitous Computing as noted by Mark Weiser in his famous essay envisioned environments with hundreds of invisible computing devices that are able to provide various services to a user [25]. Recent advances in processing power allow the integration of low-power, high-performance systems in small form factors, enabling computing devices that meet the demands of ubiquitous systems.

Interaction with intelligent environments should be based on intuitive and natural interaction metaphors, for example the interaction via speech and gesture [22]. In the area of gesture recognition, capacitive sensors allow detecting the presence of objects, such as fingers and hands, and are therefore commonly used in touch screen devices to register multi-finger input [1]. The less widely used variant of capacitive proximity sensing allows the recognition of objects at greater distances without requiring touch. Depending on their configuration, these sensors enable us to detect the presence of a body part in a proximity of some centimeters up to more than one meter [17].

The first example of this technology - a musical instrument called Theremin - dates back almost a century [10]. More recently capacitive proximity sensors have been a research interest of human computer interaction groups who investigate many applications in hand and body tracking [28, 27, 3, 12, 23]. Considering smart environments, it becomes apparent that capacitive proximity sensors are particularly well-suited for realizing unobtrusive interaction systems. The generated electric fields are only partially disturbed by non-conductive materials, such as wood, glass or concrete [2]. Therefore, this type of sensor can be hidden easily in the environment, for example under tables or in floors. However, noise and ambiguity of sensor readings is a common problem for those systems making it difficult to infer high level information from raw and unfiltered sensor data.

In a typical scenario we would like to infer object characteristics depending on the current application. This may range from reconstructing hand positions in explicit interaction devices to recognizing whole-body parameters in intelligent furniture. In order to infer those parameters, various approaches were realized that rely on the classification of capacitive sensor data. In [19] the authors have classified the way people interact with everyday objects. Cohn et al. [6, 7] overcome weaknesses of a limited detection range by applying wearable capacitive sensors, which are used to classify a discrete position of a person within a room. Moreover, capacitive sensors have been used to recognize persons by measuring their capacitive fingerprint [13]. While all these works employ sophisticated classification approaches, there are only few methods for recognizing continuous object parameters using capacitive sensors. The extraction of such parameters typically follows simplified techniques relying on strict assumptions about the type of objects to be recognized, at the cost of not being easily applicable to different objects [21]. The commercially available Cypress TrueTouch technology employs a 2.5-dimensional object recognition method for tracking the position of fingers [8]. The object recognition method

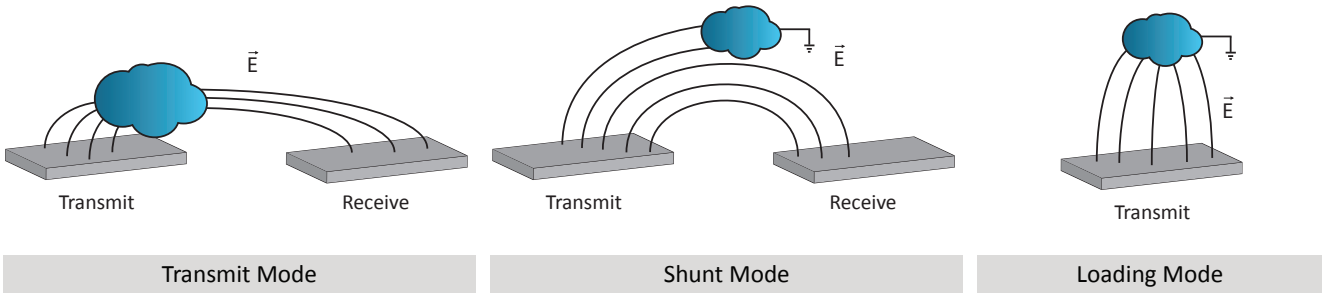

Figure 1. Measurement modes for capacitive proximity sensing.

enables to distinguish between touch and hover actions. A generic method for recognizing continuous object parameters will simplify such adaptations and provide increased flexibility.

One example of a generic method is the Swiss-Cheese-Algorithm, proposed as future work by Smith et al., that can be applied to infer an image of the surrounding environment combining the knowledge of many capacitive proximity sensors [20]. Based on this briefly introduced idea, we have formulated a novel method to recognize and track multiple objects. The final outcomes of the algorithm are configurations of body parts that can be tracked in real-time. To test the proposed method we have built a custom sensing array and applied it to the task of multi-hand gesture recognition.

In summary we provide the following contributions:

1. We present a generic object recognition and tracking method based on the Swiss-Cheese-Algorithm [20]
2. The method is implemented on a custom designed multi-hand interaction system
3. We evaluate the presented method and compare it to current multi-touch technologies

OBJECT RECOGNITION USING CAPACITIVE PROXIMITY SENSORS

Capacitive proximity sensors can be designed using different measurement modes, as illustrated in Figure 1 [28]. The transmit mode relies on a changing electric potential that is coupled with a person's body. This coupling turns the user into a transmitter whose signal can be picked up by one or more receivers. Researchers have applied this mode to identify users on multi-touch tables [9] or realized localization systems [23]. Shunt mode applies two electrodes - a distinct transmitter and a receiver. This mode can be used in combination with different multiplexing methods allowing a parallel access of many transmitters at the same time. Using shunt mode, it is possible to create numerous virtual sensors that are located in the center between each receiver-transmitter combination [11]. As this method allows a high number of measurements while having a manageable number of electrodes, it was implemented as the basis of our exemplary gesture-recognition system. Loading mode uses a single electrode that creates an electric field without using an explicit receiver. The electrode directly builds up an electric field with objects in the environments. Due to its simplicity, this mode was adopted for realizing larger activity recognition systems [27, 26, 12].

Inferring different object parameters from sensor data is a complex task. An exact solution would require solving electric field equations for multiple objects and electrodes. This calculation is too time-consuming for real-time calculations in embedded systems and requires including numerous detailed environmental parameters. Another prevalent issue of capacitive proximity sensors is related to a certain ambiguity in sensor readings. Considering a single sensor and its generated electric field, a small object that is close to the sensor might result in the same reading as a larger object at an increased distance [2]. Thus, a model is required that approximates the behaviour and influence of objects within an electric field. There are various practical solutions to build such a model. Typically the actual shape of the desired object is approximated by simple geometric shapes that are easier to process, e.g. spheres for modeling hands or cylinders for modeling arms [21]. Reducing the complexity even further they are often considered uniform in size and shape which allows associating sensor values to a specific distance [3]. However, reducing the number of parameters reduces available information accordingly. When considering more complex scenarios, such as multi-hand gesture interaction or posture detection on furniture it becomes necessary to handle objects with multiple degrees-of-freedom and objects that are linked together with various geometric constraints. Therefore a method is required that considers these restrictions and allows recognizing and tracking the state of various objects in real-time.

SWISS-CHEESE EXTENDED

The basic idea of the Swiss-Cheese-Algorithm is to detect objects using elimination [20]. Initially it is assumed that the objects may be located at any position in the interaction space. Based on the measurements of each sensor we can make assumptions about the space in which no object may exist, reducing the probability of object presence around a certain proximity to the sensor. Combining the readings from many sensors we end up with a structure not unlike a Swiss cheese, with regions that may contain an object and others that are distinctively empty. While the basic idea of this algorithm has been outlined in the past as an outlook on future work, there has not yet been any concrete implementation or theoretical formulation [21]. In the following, we present a conceptual

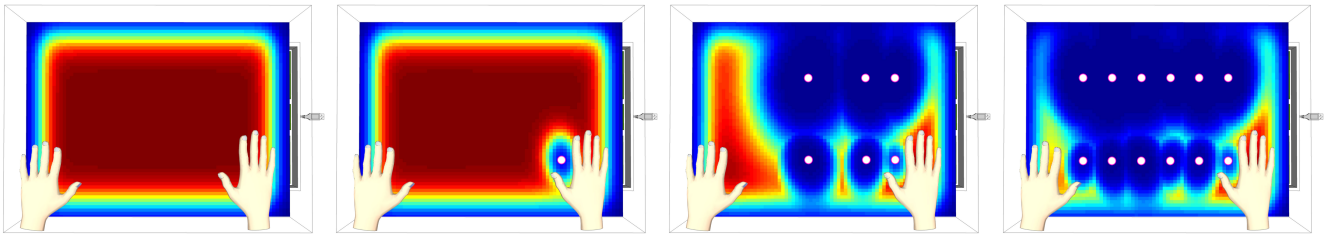

Figure 2. Swiss-Cheese-Algorithm combining the knowledge of 0, 1, 6 and 12 sensors to recognize two hands. The figure shows a 2-dimensional layer of the Swiss-Cheese-Algorithm's outcome directly underneath both hands. White dots denote the center of an active sensor (receiver-transmitter combination). Red colors denote high probability of object presence (close to 1), while blue colors denote low probability of object presence (close to 0).

and mathematical foundation of the Swiss-Cheese-Algorithm and various extensions that facilitate object recognition and tracking.

Method

In this subsection, we give a short overview about the processing steps of our object recognition and tracking method. The method is feasible for many different application scenarios and can be easily adapted. We illustrate these steps with our study of a multi-hand gesture recognition device, shown in Figure 3.

We aim to determine the most likely configuration of body parts based on the readings of many distributed proximity sensors. One important requirement of the method is the applicability on environments where it is not feasible to deploy a large amount of sensors. Thus, we have to make preliminary considerations about the recognizable objects and their degrees of freedom. As a first step we define a volumetric model of the object to be recognized. Referring to our study of a multi-hand interaction device that is shown in Figure 3, we aim to recognize the 3D-positions and grabbing state of one or two hands. Therefore, the hands are modeled as boxes with a variable x/y-edge length and an (x,y,z)-position, resulting in a 5-dimensional descriptor, the *object state*. While the position of the center-of-gravity of this box is directly associated to the position of the hand, the edge length in two dimensions and their ratio to each other are used as indicator for the grabbing state.

In the algorithm's first execution step, a volumetric object is defined that encloses the whole interaction space, that we are calling cheese. This cheese can be regarded as a 3-dimensional pseudo probability distribution for object presence in each point [20]. At the beginning of the algorithm, the presence of body parts is considered with equal probability everywhere, comparable to a cheese without holes. The algorithm has to cope with a high degree of ambiguity as sensors might deliver the same sensor reading for varying object sizes and distances. Thus, the algorithm can only make considerations about the space around a sensor in which definitely no object is present. This space can be modeled using an ellipsoid around the sensor's center. In the following, these ellipsoids are cut out of the cheese for each sensor. The parts left over contain the objects we want to recognize.

Let us illustrate this procedure with an example shown in Figure 2. It shows a 2D-layer of the cheese that is located in the interaction space above the multi-hand interaction device presented in our study. Ellipsoids are cut out of that cheese and the position of the two hands is subsequently revealed. Afterwards, the defined volumetric models of the objects are fit into the remaining cheese to obtain a probability measure for different object part configurations. However, it is typical that a large portion of cheese remains, as the sensors are usually not able to constrain the interaction sufficiently in all directions. Thus, we associate a higher weight to the object state, when it is located closer to a sensor compared to states that are located at greater distances. Using the example of the gesture recognition device it is easy to see that if a hand is located 10 cm above the sensing plane, the probable object configurations are recognized at this distance and above.

In order to determine the most likely system states in real-time, it is not feasible to evaluate all possible object configurations. Especially when the number of targets increases or the object state vector's dimensionality is high, a systematic approach for finding the most probable object configuration has to be considered. Particle filters, also known as Sequential Monte-Carlo method, provide a solution to this problem. These filters can be incorporated to evaluate only the most probable object configurations based on a spatio-temporal relationship. Multiple objects can be tracked using separate instantiations of a particle filter, which enables us to track two hands in real-time with our multi-hand interaction device.

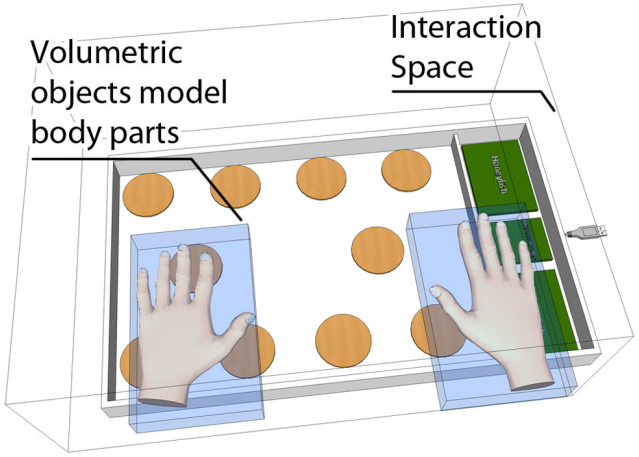

Figure 3. Our multi-hand interaction device with hands modeled as volumetric objects. 10 copper plates are used as electrodes that build up an electric field to the user's hands.

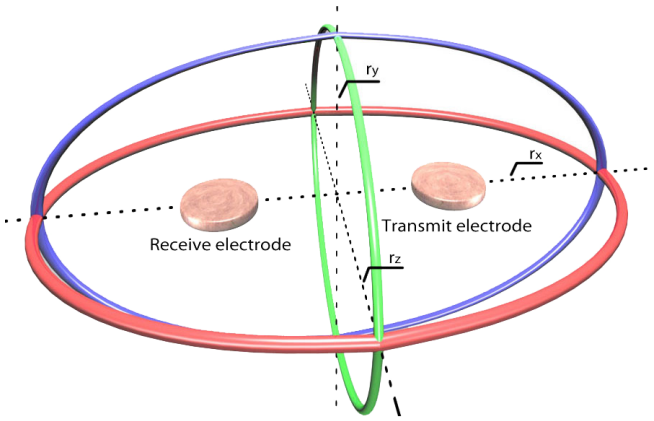

Figure 4. An ellipsoid with three independent semi-principal axes $r_{x,y,z}$ models the distance of a unit absorber to the sensor's center

Object recognition

Forward reading model

The forward reading model is used to reconstruct a sensor reading f that one would obtain when a unit absorber is placed at a location (x, y, z). A unit absorber is a very small conductor and represents the nearest possible object in the environment of a sensor for a given sensor reading. However, it is possible that bigger objects cause the same sensor reading at greater distances from the sensor. The forward reading f can be seen as a prediction of a sensor reading for an imaginary unit absorber. Later, this prediction is used for comparison with the actual sensor value.

The *iso-signal shell* is a surface around a sensor on which a unit absorber causes the same sensor reading [21]. Thus, this surface marks a volume around a sensor in which no object may be present. Due to the proximity sensor's layout with two electrodes located beside each other, a sensor's reading depends on the direction in which an object approaches. For example, a unit absorber that is placed vertically above a sensor at a distance of $10cm$ could produce the same sensor reading as a unit absorber placed horizontally aside a sensor at a distance of $15cm$. These axis-dependent characteristics can be expressed by modeling the iso-signal shell as an ellipsoid, as shown in Figure 4. This ellipsoid is composed of three independent semi-principal axis $r_{x,y,z}$. When a unit absorber is placed at a point (x, y, z), it is located on the iso-signal shell if the following ellipsoidal condition is fulfilled:

$$\frac{x^2}{r_x^2} + \frac{y^2}{r_y^2} + \frac{z^2}{r_z^2} - 1 = 0 \tag{1}$$

Depending on the applied measurement mode, a model function with different parameters for each axis is required to calculate the forward reading. This model function relates the forward reading to the distance of a unit absorber. A simplified model can be based on the electric field strength in a given object location [21]. Considering shunt mode measurements with a dipole approximation based on point charges, this means that the electric field strength around a sensor decreases with the factor d^3, related to the distance d of a unit absorber from the sensor. Using this approximation we can derive the following model function for each axis $i = x, y, z$ to model the axis-dependent directivity: [20]

$$f = 1 - \frac{1}{(\alpha_i + \beta_i r_i)^3}; i = \{x, y, z\} \tag{2}$$

$$\Leftrightarrow r_i = \frac{\beta_i}{\sqrt[3]{1-f} - \alpha_i} \tag{3}$$

The equation is composed of two fit parameters for a single axis, $\alpha_{x,y,z}$ and $\beta_{x,y,z}$. These fit parameters are applied to model the gradient of the electric field strength along the given axis. The fit parameters can be determined experimentally by moving a unit absorber along an axis and recording the sensor value in relation to the unit absorber's distance to the sensor. In the following step, the fit parameters are calculated using least-squares fitting.

We now combine the given fit function with the ellipsoidal condition to determine the forward reading for a unit absorber located at a certain point (x, y, z) on the iso-signal shell. Since the three semi-principal axis $r_{x,y,z}$ are unknown, we can replace them with the determined fit function given in Equation 3. We yield an equation with just one unknown variable, the forward reading f:

$$\left(\frac{x}{\frac{\beta_x}{\sqrt[3]{1-f}-\alpha_x}}\right)^2 + \left(\frac{y}{\frac{\beta_y}{\sqrt[3]{1-f}-\alpha_y}}\right)^2 + \left(\frac{z}{\frac{\beta_z}{\sqrt[3]{1-f}-\alpha_z}}\right)^2 - 1 = 0 \tag{4}$$

The equation cannot be resolved analytically to f, the result would be a polynomial of order 6. Thus, we choose to solve Equation 4 by minimization using the variable forward reading f. As a normalized sensor reading is restricted to a range of $[0, 1]$, the value of f can be determined efficiently with methods like the Brent algorithm [4] in real-time. As an outcome of the minimization approach we can now determine a forward reading f each sensor would produce when a unit absorber is located at a given point.

Prediction of object presence

We can only make considerations about the space around a sensor in which no absorber can be located. This space is limited to the distance between a unit absorber and a sensor that can be regarded as the nearest possible object. In this step, spaces in which no object may be located are cut out of the pseudo probability distribution. To evaluate a point $\vec{p}$ in space, the forward reading f is subtracted from the actual sensor reading s that was measured, resulting in $\delta = f - s$.

To emphasize the meaning of the value δ, consider a unit absorber located $10cm$ above a sensor that would cause a normalized sensor reading $s = 0.5$. Applying the forward reading model for an imaginary unit absorber in a distance of $5cm$ above the sensor would yield a forward reading of $f = 0.2$.

When the distance of the imaginary object comes closer to $10cm$, a forward reading of $f = 0.5$ would be determined. At greater distances, for example $15cm$, one would yield a forward reading of $f = 0.8$. When computing the difference $\delta = f - s$ of the actual reading s and the forward reading f, we can conclude that no object can be present for $\delta < 0$ and an object can be present for $\delta \geq 0$. However, based on this assumption one can only conclude that the nearest possible object is located at 10cm above the sensor, but it is also possible that a bigger object is located at a distance of $11cm$ or beyond.

The difference value δ is an input argument to a sigmoidal function, with two parameters μ_n (displacement) and γ_n (steepness) where $n = 1, 2, ..., N$ denotes the sensor:

$$P_n(\vec{p}) = \frac{1}{1 + e^{(-\gamma_n \cdot (\delta - \mu_n))}} \tag{5}$$

The function expresses that the probability of an object being within the inner region of an iso-signal-shell of a unit absorber is close to zero. In the space outside the iso-signal-shell, the prediction is close to one. When the steepness γ_e converges to infinity, then the sigmoidal function can be regarded as a simple Heaviside function. Lower values for that parameter can alleviate the effect of noisy measurements by expressing a level of uncertainty. In the following, the knowledge gathered from all sensors is combined:

$$P(\vec{p}) = \prod_{i=1}^{N} P_n(\vec{p}) \tag{6}$$

Thus, when all sensors are sure that an object may be present at a given point, the function $P(\vec{p})$ will evaluate close to one. If the point is within a space in which one or multiple sensors do not consider an absorber, the function evaluates close to zero.

Body part representation

Based on the previous findings, we are able to obtain a measure for object presence in a single point. In the following, an approach for determining the state of an object is presented. As explained in the overview of this section, an object state can be embodied by a location and the properties of a volumetric object. However, it is possible to employ more complex geometrical models that are composed of many volumes and must be described by a higher number of parameters. It is necessary to find a suitable compromise between the object's shape and the accuracy of the model. It is not viable to apply a fine-grained arm model for the distinction of different fingers if the required information is not contained in the sensor data.

The most probable object state can be determined by maximizing the average volume integral over the pseudo probability distribution in each point that is enclosed by the volumetric model. The volume integral over the function $P(\vec{p})$ is solved using a Monte-Carlo integration. V denotes the object's volume, M the number of Monte-Carlo samples, and $\vec{p_i}$ a sampling point within the object:

$$\iiint_V P(\vec{p}) d\vec{p} \approx V \cdot \frac{1}{M} \cdot \sum_{i=1}^{M} P(\vec{p_i}) \tag{7}$$

To obtain the Monte-Carlo integral of a function, a uniformly distributed set of points within the volume must be determined. For each of these points, the prediction of object presence is computed. With an increasing number of points, the error between the actual integral and the Monte-Carlo integral is minimized. However, a high number of points leads to computationally higher cost.

In order to obtain meaningful results, the object state parameters must be limited in a way that restricts the object position to the interaction space and the object shape to feasible variants. As the interaction space is usually not restrained by sensors in all directions, the object configurations that are closer to sensors must be weighted higher as object configurations that are far away from the sensors. This linear weighting can be accomplished by calculating the distance to the nearest sensor or the distance to a sensing surface, when the sensors are located in a plane.

Object Tracking

In the previous section, a method for object recognition was introduced. Using the Swiss-Cheese-Algorithm, it is possible to obtain a measure of probability for object presence in each point in space. This is the basis for the recognition of objects that can be modeled by basic geometric shapes, such as a box. The goal of object tracking is to estimate a system state, employing a set of measurements in real-time. This estimation does not only depend on the current time-step, but also on the system state's evolution in time. In single-target tracking, a system state can be expressed by a single object state, for example by the position of hand modeled by a box. When more than one object is tracked, the system state is the combination of all distinct object states. A *system model* incorporates the change of a system in time, whereas the *measurement model* is utilized to evaluate the probability of a hypotheses [14].

In order to determine the most likely system states, it is not feasible to evaluate all possible system states in real-time. Especially when the number of targets increases or the object state vector's dimensionality is high, a systematic approach for finding the most probable system state has to be considered. Particle filters reveal their strengths in the possibility to track many hypothesis about an object state in a spatio-temporal relationship. The concept makes them robust against occlusion and clutter [14]. This robustness can be exploited in capacitive proximity sensing, as measurement noise and fast movements pose comparable challenges on the filter. Moreover, maintaining these spatio-temporal relationships can enhance the recognition rate when objects leave the interaction area for a short time.

Tracking multiple targets poses various challenges on particle filtering. Standard particle filters are not suited for track-

ing a varying number of targets. In particular, the samples quickly converge to a single target when more than one target is present [24]. To overcome this limitation, many extensions to particle filters were proposed [15, 24]. When the number of targets T is known in advance, it is possible to represent the system state as a joint set of object states [24]. Problems arise when the object states become more complex and the number of targets increases. In this case, the dimensionality of the system state vector quickly becomes unhandy and the system performance decreases.

In order to avoid a rising complexity with an increasing system state dimensionality, targets can be tracked independently with separate instantiations of a particle filter. It is essential to protect samples that represent local maxima of the probability distribution from extinction. Milstein et al. present the idea of a clustered particle filter [18], that inspired our multi-target tracking approach. In each step, the determined probability distribution is clustered for a variable number of targets. For each cluster, a fixed number of samples is selected for the next sampling stage. In each step, the number of targets is determined from the variance of object states within a corresponding cluster. When the variance is high, the number of targets is increased and the clustering process is repeated. When no good object states can be found within a cluster, the number of targets is decreased.

Furthermore, the task of tracking newly appearing targets is not considered in standard particle filtering [15]. When particles track an existing target, a newly appearing target can only be recognized by particles that migrate from the existing target to the new one. In order to solve this problem, we determine an initialization density directly from the sensor readings. When a sensor yields a reading that indicates a nearby target, particles with expected object states are randomly initialized in the neighborhood of that sensor. In each initialization phase, a fixed number of particles is distributed over the state space.

Target management

The outcomes of the particle filtering approach for multi-target tracking are cluster centers that represent recognized targets and their properties. Target management is the task of keeping track of a uniquely identifiable target object through time and handling newly appearing and vanishing targets. Thus, the determined cluster centers that exceed an observation threshold are connected to a set of maintained target objects. A target object has a unique ID, a history vector of all identified cluster centers and threshold values that are used to compensate noisy detections and to maintain non-detected targets through measurement noise. For each time step, the target management assigns the recognized cluster centers to a set of maintained target objects. Therefore, the distances from the new cluster centers to the last assigned cluster center of all target objects is calculated. Then, the nearest cluster centers are assigned to the existing target objects. If more cluster centers than existing targets are recognized, the remaining unconnected cluster centers are used to create new target objects. When less cluster centers are recognized, targets that were not assigned to a cluster center are removed.

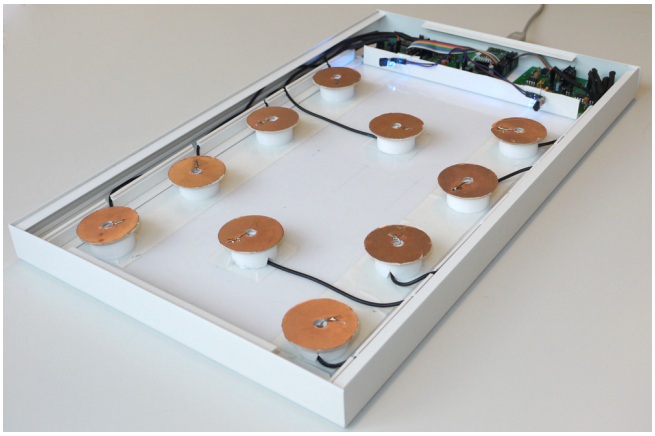

Figure 5. The prototypical multi-hand interaction device employing shunt mode measurements. Two receivers are placed in the center while 8 transmitters are located at the device's edges.

Interpolation

In general, the recognized cluster centers have relatively smooth trajectories over time. However, there might be variations due to noisy measurements and the probabilistic nature of particle filters. Thus, interpolation techniques can improve the continuity of trajectories in applications that track gestures. Moving average filtering of a target object's past cluster centers can smooth the trajectories and lead to higher precision. A moving average a for a target object with a history h of past cluster centers with window size L can be determined as follows:

$$a = \frac{1}{L} \sum_{i=0}^{L} h_{t-i} \tag{8}$$

An averaging approach with a fixed window size L faces the great disadvantage of increasing the latency to an unacceptable amount. When having smaller window sizes, the system's reaction time increases whereas the smoothness decreases. Most gesture-recognizing applications require low precision and latency while fast movements are performed. For tiny movements, the precision is considered to be more important than latency. Thus, we apply an adaptive moving average filter that determines the size of the input history cluster centers depending on the object's movement speed.

STUDY: GESTURE RECOGNITION DEVICE

Prototype

We created a hardware platform for gesture recognition that is shown in Figure 5 [11]. The platform operates in shunt mode and applies a combination of time-division and frequency-division multiplexing for parallel transmitter operation. The gesture recognition prototype uses two synchronized boards, each driving four transmitters and one receiver. The boards are able to receive transmitted signals from each other and can be easily extended. We transmit frequencies of 10-25KHz, sample the received signals at 100KHz and apply

a Fast Fourier Transform for reconstruction of the transmitted signal amplitude. The transmitted sine-wave signals have a peak-to-peak amplitude of 5V. The multiplexing approach enables us to retrieve 50 samples per second for each receiver-transmitter combination. We created a hardware platform for gesture recognition that is shown in Figure 5 [11]. The platform operates in shunt mode and applies a combination of time-division and frequency-division multiplexing for parallel transmitter operation. The gesture recognition prototype uses two synchronized boards, each driving four transmitters and one receiver. The boards are able to receive transmitted signals from each other and can be easily extended. We transmit frequencies of 10-25KHz, sample the received signals at 100KHz and apply a Fast Fourier Transform for reconstruction of the transmitted signal amplitude. The transmitted sine-wave signals have a peak-to-peak amplitude of 5V. The multiplexing approach enables us to retrieve 50 samples per second for each receiver-transmitter combination.

The gesture recognition device consists of eight transmit electrodes located at both sides of the device and two receive electrodes that are placed in the center of the sensing plane. This mode of operation makes it possible to use 16 virtual sensors, one virtual sensor per receiver-transmitter combination. Applying this setup, we can detect fast multi-hand gestures above an area of 40 x 20cm with a maximum detection height of approximately 20cm.

Supported Gestures

Discrete gestures represent actions that trigger a discrete command, such as page turning. The recognition is based on the covered distance and movement direction of a target object, whereas the movement speed is of secondary interest. Thus, the movement history of each target object must be analyzed continuously. This processing takes place in each time-step applying a sliding window on the history of object states.

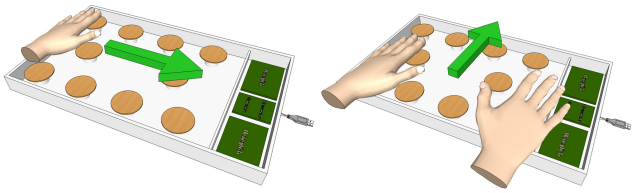

Figure 6. Swipe gestures from left to right with a single hand and from bottom to top with two hands

Swipe gestures, visualized in Figure 6, are well known in multi-touch applications and are often applied on image browsing or changing views [16]. This gesture type can be performed with a single target, but is also applicable on two targets moving in parallel. A swipe gesture is based on a movement parallel to a reference axis with low deviations to the orthogonal axis. Furthermore, it needs to be performed with a certain movement velocity. An average velocity in the x- and y-direction and the movement distance is calculated for a window with its past cluster centers. When the velocity in a single direction and the distance exceeds a threshold, a swiping action is recognized. In order to properly recognize a swiping action, the average velocity to the orthogonal direction must lie within an error threshold.

Continuous zoom and rotation gestures with two hands are shown in Figure 7 [5]. They are analogous to pinch/zoom and rotate gestures known from multi-touch applications [16]. In contrast to multi-touch, gestures are not performed using two fingers but with two hands. As soon as two hands are recognized, the corresponding angle between the hands with respect to the device's longitudinal axis is calculated, representing the desired rotation. The distance between both hands is mapped to a zoom factor.

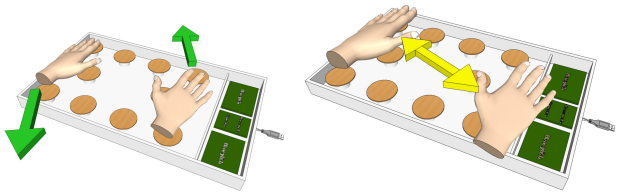

Figure 7. Combined zoom and rotation gesture (green) and the corresponding zoom and rotation axis (yellow)

A grasp action, depicted in Figure 8, can be used for drag-and-drop gestures or to activate a different gesture interaction set. For example, it is feasible to perform a grasp gesture in combination with a swipe gesture to control different parts of an application. Grasp actions can be recognized depending on the object's length and width.

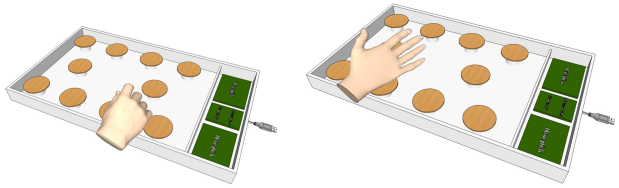

Figure 8. Grasp and release actions can be utilized for drag-and-drop functionality

In contrast to pure multi-touch applications, variants employing capacitive proximity sensing have certain limitations regarding direct interaction. Considering multi-touch applications, the interaction barriers are always apparent: interaction starts when a finger touches the multi-touch surface and ends when the finger is removed. Regarding capacitive proximity sensing, the user can only make preliminary considerations about the interaction barriers, for example the height in which a hand may be detected. Thus, the interaction barriers are fuzzy and not apparent to the user.

Due to this important fact, direct feedback on the interaction status is very helpful for a user. Furthermore, it is necessary to soften Boolean decisions like selection actions. An activation state indicates when some predefined constraints are not or only partly fulfilled. These constraints can employ the vertical hand distance or a timer-controlled activation delay. Such a delay can be used for the object selection and triggering of region-dependent actions. An exemplary activation state feedback was realized using a vertical timer bar underneath each cursor. When the cursor is not able to trigger an action,

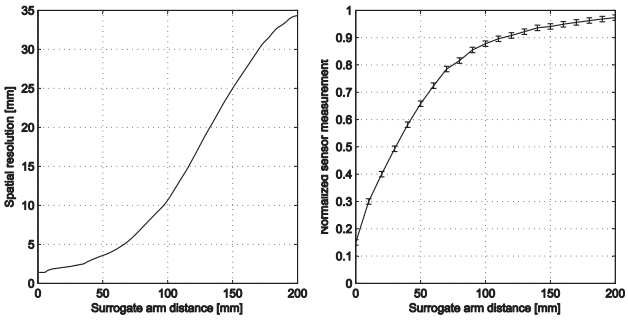

Figure 9. The left plot shows a sensor's resolution which decreases with higher object distances. The normalized sensor values and their standard variance for a constant surrogate arm distance are shown in the right plot.

this means it is passive, it is marked in white. An active cursor, indicating that gestures can cause immediate actions, is marked in blue. Beside these gestures, region-dependent gestures were realized that trigger commands as soon as a hand remains over a predefined region. For example, this gesture type can be utilized for implementing continuous scrolling behaviors in an application. Region-dependent gestures can also be protected by a timing threshold, such that accidental movements do not cause immediate effects. Moreover, the time that a hand rests above such a region can be employed to continuously effect interaction properties, for example increase the scrolling speed with respect to the resting time.

Evaluation

Object recognition and temporal performance

The object recognition performance highly depends on the hardware being used. In order to characterize the object recognition performance, we adopted a measurement setup proposed by Smith et al. [21], which was later used by Wimmer et al. [27]. The test setup uses a grounded aluminum tube acting as a surrogate arm. We took the vertical distance to the capacitive proximity sensors in relation to the acquired sensor values with their standard variance as a measure for the system's resolution. Assuming Gaussian noise, the resolution expresses that the reconstructed distance is in a given range for 68.3% of all acquired sensor values. Figure 9 shows the normalized sensor values with their standard variance for the given vertical distance of the surrogate arm. In our evaluation we achieved a resolution of approximately 3.5mm at object distances around 50mm, and 35mm at object distances of 200mm. The resolution plot indicates that the resolution decreases with the vertical distance of an arm. This property of a capacitive proximity sensing system can be expressed in the object recognition model with the steepness factors γ_n.

The processing chain introduces several temporal delays caused by the hardware, particle filtering, target management, and interpolation. While the capacitive sensing hardware is able to measure the proximity to an object in realtime, the worst-case sensor update rate and the PC communication introduce a delay of approximately 30ms. With a particle filter update rate of 25Hz, new objects are usually recognized in 1-2 particle filter iterations. To suppress noisy detections at the border of the device, the target management introduces an additional delay by including new targets only after two succeeding detections. This results in a total delay of 150ms for newly appearing objects. Existing objects are tracked with delays mainly resulting from interpolation. For small movements in the area of a few millimeters, this delay is approximately 150ms in total, too, averaging over 3 succeeding object configurations. For fast movements in the area of 20cm, the total delay is only 70ms as the adaptive averaging interpolation is limited to a single object configuration, thus removing the delays associated with interpolation.

Usability evaluation

A usability evaluation was conducted at the student fair Hobit in Darmstadt (shown in Figure 10) with 18 participants, the majority not having a technical background. The evaluation's goal was to obtain feedback on the general user experience, including precision and reaction time, evaluate suitable applications and to compare the prototype's performance to a multi-touch system. As a prerequisite, a short introduction into the technology and the handling of the applications was given to every participant.

The first part of the evaluation focused on two gesture-controllable applications: an image viewer and a gaming application[1]. The prototype was placed on a table in front of a screen showing the application. The participants could choose to either sit or stand during the evaluation. Regarding the image viewer, each person had to accomplish a predefined set of tasks, such as image rotation, selection and browsing. The gaming application was evaluated four times - single-handed and two-handed - with one repetition to assess the learning curve and determine if users favor either multi-hand or single-hand interaction. The collected points as well as the total time to finish a game level was recorded. In order to compare a participant's experiences with the 3D-interaction approach to a multi-touch enabled device, the same tasks had to be conducted on an ACER Iconia tablet running a standard image gallery based on Android. The reason for this comparison is the extendibility of multi-touch devices based on capacitive sensing to register 3D-interaction using the same

[1]Tux Racer - tuxracer.sourceforge.net

Figure 10. Usability evaluation at the student fair Hobit in Darmstadt. The electrodes are hidden under a surface made of acrylic glass.

technique. Therefore, our object recognition method can be applied as a generalized approach for interacting above a capacitive sensing device. In the second part of the evaluation, the users were asked to fill out a questionnaire to provide some qualitative feedback about their experience. The subjects had to rate their experiences on a Likert scale from 1 (no approval) to 10 (full approval). Additionally, they were asked to identify future application scenarios and advantages/disadvantages compared to multi-touch technologies.

Many test subjects experienced the evaluated prototype and its applications to be intuitive (8.71 approval) and uncomplicated. Most of them had the impression that the provided tasks could be accomplished easily (7.47 approval) and the system's reaction was comprehensive (6.94). Almost all subjects could imagine using a similar interaction device on a regular basis (8.53 approval) and deemed that the evaluated prototype is an interesting interaction modality (9.59 approval). They were fascinated by the possibility of contactless and gesture-based interaction. The large size of the prototype's interaction area was also a compelling factor. Gestures such as swiping were experienced to be recognized fast and with great precision. Both evaluated applications, the image viewer (8.0 approval) and the gaming application (8.59 approval) appealed to many subjects. Regarding the gaming application, the subjects were able to rate their favorite interaction mode on a Likert scale from 1 (single-hand) to 10 (dual-hand). Most users were either attracted to single-hand or dual-hand control, only few users liked both interaction modes. The gaming application, that was evaluated twice for each interaction mode, showed that there is a flat learning curve, letting users master the game quickly. Many users did not improve during the two rounds and achieved equally good scores from the beginning on.

The subjects had problems with the system's reaction time (5.88 approval to very fast recognition compared to very slow recognition), that can reach a maximum latency of 150ms. Especially in the gaming application, this latency turned out to be critical. Moreover, some people identified the lack of precision (5.59 approval to very high precision compared to very low precision) as an unpleasing factor. A few subjects criticized the system's recognition bounds that were not marked explicitly. Furthermore, the test persons experienced a tiring interaction posture that was caused by low table height and stretched arms. In many cases, the subjects unsuccessfully tried to influence the cursor position relatively to the current position, in a similar way as using a normal touchpad.

Compared to multi-touch interaction, the obvious advantage of contactless interaction was stated out by many people. Moreover, 3D-interaction can offer more modes of interaction. The subjects mentioned advantages like easier interaction (less fine-grained) and a seamless and invisible integration into the environment. Interaction can also be performed when wearing gloves and with less attention. On the other hand, the test persons stated that multi-touch is more precise and faster. In a direct comparison between a multi-touch image viewer and the gesture based image viewer, the multi-touch image viewer was favored by most people. This can be explained by the high interaction speed and precision that can be achieved with multi-touch technology.

The test persons were asked to identify future application scenarios. Most test persons could envision systems using capacitive proximity sensing in the area of home entertainment. In particular, the subjects suggest controlling TVs, audio, game consoles and personal computers using this technology. Due to the contactless interaction, medical and surgical applications were also mentioned very often. In contrast to multi-touch technologies, such systems might offer great advantages regarding sterility and cleanability. This is a major concern of many users who identified applications in public transport (ticket machines) and public sanitary installations. The contactless and invisible integration in furniture or behind walls and doors is an important advantage for many users. Seniors who are not able to perform fine-grained movements, for example those suffering from Parkinson's disease, can benefit from systems that recognize coarse gestures. Furthermore, conference rooms and presentation environments were proposed to be equipped with gesture recognizing systems that facilitate the interaction with such a complex technical environment.

Summing up, almost all participants liked contactless gesture-based interaction and experienced it to be intuitive and comprehensive. The precision and reaction time were criticized by several subjects. We plan to improve the precision by using a higher number of sensors and experiment with different volumetric representations. The latency is caused by a Java implementation running on a PC, which is currently migrated to the interaction device's microcontroller. In contrast to multi-touch technology, the interaction speed with an application is significantly slower due to longer lasting gestures. However, the strength of contactless gesture recognition lies in different application fields that cannot be covered by multi-touch technology.

DISCUSSION AND CONCLUSION

In this paper we presented *Swiss-Cheese Extended*, a novel method for recognizing and tracking objects in ubiquitous user interfaces using arrays of capacitive proximity sensors. We formulated and implemented the Swiss-Cheese-Algorithm that allows generating a measure for object presence in space. The algorithm was extended by application-dependent volumetric object representations and by adding customized particle filtering to track existing and newly appearing objects in real-time.

In order to evaluate our method we created a prototype for multi-hand gestural interaction. Based on this custom-built system the presented method allows to reliably track the state of two hands using just 16 sensor channels. Our system supports various gestures, including multi-hand rotation and zoom as well as grasping actions. These gestures allow controlling different demonstration applications that were adapted for our input device. To evaluate the performance and user experience, we finally performed a study that compared our system to a multi-touch tablet. While the test

persons considered the system improvable in terms of interaction latency and precision, most participants valued the intuitiveness and novelty of the device. Various potential application scenarios were identified, ranging from medical solutions, where sterility is crucial, to unobtrusive integration in furniture. Particularly in assistive applications, the presented method can enrich the execution context, for example by identifying unhealthy postures in beds.

As a first future step, the sensor system is currently optimized to improve the interaction speed and precision, being the main points of criticism in our study. Future iterations will outsource the processing steps of object recognition and tracking to a dedicated microcontroller. We plan to apply our object recognition approach to the domain of whole-body interaction and estimate body poses using a human skeleton model. Finally we intend to investigate additional application scenarios with different measurement modes, electrode materials and shapes.

Acknowledgements
We would like to thank the visitors of the student fair Hobit for taking part in our evaluation.

REFERENCES

1. Barrett, G., and Omote, R. Projected-Capacitive Touch Technology. *Information Display 3*, 26 (2010), 16–21.
2. Baxter, L. K. Capacitive Sensors: Design and Applications. In *IEEE Press Series on Electronics Technology* (1997).
3. Braun, A., and Hamisu, P. Designing a multi-purpose capacitive proximity sensing input device. *PETRA '11* (2011), 151–158.
4. Brent, R. Algorithms for Minimization Without Derivatives. In *Dover Publications* (2002).
5. Buxton, B., and Myers, B. A. A study in two-handed input. In *CHI '86* (1986), 321–326.
6. Cohn, G., Morris, D., Patel, S., and Tan, D. Humantenna: using the body as an antenna for real-time whole-body interaction. In *CHI '12* (2012), 1901–1910.
7. Cohn, G., Morris, D., Patel, S. N., and Tan, D. S. Your noise is my command: sensing gestures using the body as an antenna. In *CHI '11* (2011), 791–800.
8. Cypress Semiconductor Corp. Cypress TrueTouch Touchscreen Solution Drives "Floating Touch" Navigation Feature in New Xperia sola Smartphone from Sony Mobile Communications, 2012. *Press release, http://www.cypress.com/?rID=60561 (accessed 12/17/2012)*.
9. Dietz, P., and Leigh, D. DiamondTouch: A Multi-User Touch Technology. In *UIST '01* (2001), 219–226.
10. Glinsky, A. Theremin: Ether Music and Espionage. In *University of Illinois Press* (2000).
11. Grosse-Puppendahl, T., and Braun, A. Honeyfish - a high resolution gesture recognition system based on capacitive proximity sensing. In *Embedded World Conference '12* (2012).
12. Grosse-Puppendahl, T., Marinc, A., and Braun, A. Classification of User Postures with Capacitive Proximity Sensors in AAL-Environments. In *AmI '11* (2011), 314–323.
13. Harrison, C., Sato, M., and Poupyrev, I. Capacitive fingerprinting: exploring user differentiation by sensing electrical properties of the human body. In *UIST '12* (2012), 537–544.
14. Isard, M., and Blake, A. Condensation — conditional density propagation for visual tracking. *International Journal of Computer Vision* (1998), 5–28.
15. Koller-Meier, E. B., and Ade, F. Tracking multiple objects using the Condensation algorithm. *Robotics and Autonomous Systems 34*, 2-3 (2001), 93–105.
16. Kurtenbach, G., Fitzmaurice, G., Baudel, T., Buxton, B., and East, R. S. The Design of a GUI Paradigm based on Tablets, Two-hands, and Transparency. In *CHI '97* (1997), 35–42.
17. MacLachlan, R. Spread Spectrum Capacitive Proximity Sensor, http://humancond.org/ (date accessed: 09/11/2012), 2004.
18. Milstein, A., Sánchez, J. N., and Williamson, E. T. Robust Global Localization Using Clustered Particle Filtering. *Artificial Intelligence '02* (2002), 581–586.
19. Sato, M., Poupyrev, I., and Harrison, C. Touché: enhancing touch interaction on humans, screens, liquids, and everyday objects. In *CHI '12* (2012), 483–492.
20. Smith, J. R. Field mice: Extracting hand geometry from electric field measurements. *IBM Systems Journal 35*, 3-4 (1996), 587–608.
21. Smith, J. R., Gershenfeld, N., and Benton, S. A. Electric Field Imaging. *PhD Thesis* (1999).
22. Valli, A. The Design of Natural Interaction. *Multimedia Tools and Applications 38*, 3 (2006), 295–305.
23. Valtonen, M., Mäentausta, J., and Vanhala, J. TileTrack : Capacitive Human Tracking Using Floor Tiles. In *PerCom '09* (2009), 1–10.
24. Vermaak, J., Doucet, A., and Patrick, P. Maintaining Multi-Modality through Mixture Tracking. In *ICCV '03* (2003), 1110–1118.
25. Weiser, M. The computer for the 21st century. *Scientific American* (1991), 66–75.
26. Wimmer, R., and England, D. Capacitive Sensors for Whole Body Interaction. *Whole Body Interaction* (2011).
27. Wimmer, R., Kranz, M., Boring, S., and Schmidt, A. A Capacitive Sensing Toolkit for Pervasive Activity Detection and Recognition. *PerCom'07* (2007), 171–180.
28. Zimmerman, T. G., Smith, J. R., Paradiso, J. a., Allport, D., and Gershenfeld, N. Applying electric field sensing to human-computer interfaces. *CHI '95* (1995), 280–287.