Title of the Invention: Apparatus and method for determining an intended target
Abstract Title: Determining an intended target

Human machine interaction (HMI) via gesture pointing actions with particular relevance to use within a moving vehicle, determining an intended target of an object (e.g. a finger or stylus). A location of the object is determined at time intervals, and a metric associated with each of the targets. The metric is indicative of a respective target being the intended target of the object. The metric is determined based upon a model and the location of the object at the plurality of time intervals, and may determine the trajectory of the object. A Bayesian reasoning process is used to determine the intended target from the plurality of targets based on the metric associated with each of the plurality of targets. The intended target may be an item on a graphical user interface (GUI) or a physical control. The model may be a Bayesian intentionality prediction model, nearest neighbour model, linear or nonlinear destination reversion model, an equilibrium reverting model or mean reverting diffusion model. Environmental information (e.g. indicating the acceleration of an associated vehicle) may also be received and used as in the model.

Fig. 4
(a) Vehicle is stationary.

(b) Vehicle is moving at varying speeds and over uneven road.

Fig. 1
Apparatus and Method for Determining an Intended Target

TECHNICAL FIELD

The present disclosure relates to an apparatus and a method for determining an intended target of a pointing object.

BACKGROUND

It is common for a user to interact with a machine, so called human machine interaction (HMI), via a pointing selection action, hereinafter referred to as a pointing gesture. For example the user may point to a button or other control or an interactive display such as graphical user interface (GUI) which may be displayed on a touch-sensitive display device. However, especially when such gestures are used in moving vehicles which can lead to erratic and unpredictable perturbations in the user input resulting in erroneous selection(s), this may compromise system usability and tie up an undesirable amount of the user's attention, particularly if the user is the driver of the vehicle.

It is an object of embodiments of the invention to at least mitigate one or more of the problems of the prior art. It is an object of embodiments of the invention to reduce a duration of a pointing gesture. It is an object of embodiments of the invention to improve an accuracy of a pointing gesture.

SUMMARY OF THE INVENTION

According to an aspect of the present invention there is provided a method of determining an intended target of an object, comprising determining a location of the object at a plurality of time intervals; determining a metric associated with each of a plurality of targets, the metric indicative of the respective target being the intended target of the object, wherein the metric is determined based upon a model and the location of the object at the plurality of time intervals; determining, using a Bayesian reasoning process, the intended target from the plurality of targets based on the metric associated with each of the plurality of targets.

Optionally, the intended target is determined based on the location of the object. The intended target may be determined before the object reaches the target.
The method may comprise determining a trajectory of the object. The trajectory of
the object may comprise data indicative of the location of the object at a plurality of
time intervals. Using the trajectory of the object may improve determination of the
intended target.

The method may comprise filtering the trajectory of the object. The filtering may
smooth the trajectory of the object and/or the filtering may reduce unintended
movements of the object and/or noise from the trajectory. Advantageously filtering
the trajectory may reduce an influence of unintended movements such as jumps or
jolts.

The model may be a Bayesian intentionality prediction model. The model may be a
linear model. The model may be based on one or more filters; optionally the one or
more filters are Kalman filters.

The model may be a non-linear model. The model may incorporate irregular
movements of the object. The non-linear model may be based on one or more
particle filters.

The model may be a nearest neighbour (NN) model. The NN model may determine
the metric based upon a distance between the location of the object and each of the
targets. The metric may be indicative of a distance between the object and each of
the targets.

The model may be a bearing angle (BA) model. The metric may be indicative of an
angle between the trajectory of the object and each of the targets.

The model may be a heading solid angle (HSA) model. The metric may be indicative
of a solid angle between the object and each of the targets.

The model may be a Linear Destination Reversion (LDR) or a Nonlinear Destination
Reversion (NLDR) model. The method may comprise determining a model for each
of the targets. The metric may be indicative of the model best matching the trajectory
of the object. The NLDR model may comprise non-linear perturbations of the
trajectory.
The model may be a Mean Reverting Diffusion (MRD) model. The MRD may model a location of the object as a process reverting to the intended target.

The model may be an Equilibrium Reverting Velocity (ERV) model. The metric may be based upon a speed of travel of the object to the target.

The method may comprise determining a state of the object.

The method may comprise receiving one or more items of environmental information. The environmental information may comprise one or more of information indicative of acceleration, information indicative of a state of the vehicle and/or image data indicative of vehicle surroundings. The determination of the metric may be based, at least in part, on the one or more items of environmental information. The model may be selected based, at least in part, on the one or more items of environmental information.

The determining the intended target may be based on a cost function. The cost function may impose a cost for incorrectly determining the intended target. The intended target may be determined so as to reduce the cost function.

The determining the intended target may be based on one or more items of prior information. The prior information may be associated with at least some of the targets. The prior information may be indicative of previously selected targets. Advantageously the prior information may improve determination of the intended target.

The method may comprise selecting a plurality of most recent time intervals, wherein the determining the metric associated with each of the plurality of targets may be based upon the location of the object at the plurality of most recent time intervals.

The object may be a pointing object. The location of the object may be determined in three dimensions. Determining the location of the object may comprise tracking the location of the object. Determining the location of the object may comprise receiving radiation from the object.
The method may comprise outputting an indication of the intended target. The indication of the intended target may comprise identifying the intended target; optionally the intended target may be visually identified. Advantageously the user may become aware of the determined intended target. The user may then cause selection of the intended target.

The method may comprise outputting the indication of the intended target and one or more possible targets. The method may comprise activating the intended target.

The plurality of targets may comprise one or more of graphically displayed items or physical controls. The location of the object may be determined in three-dimensions.

According to an aspect of the present invention there is provided a system for determining an intended target of an object, comprising location determining means for determining a location of the object; a memory means for storing data indicative of the location of the object at one or more instants in time; a processing means arranged to determine a metric associated with each of a plurality of targets of the respective target being the intended target of the object, wherein the metric is determined based upon a model and the location of the object at the plurality of time intervals; determine, using a Bayesian reasoning process, the intended target from the plurality of targets based on the metric associated with each of the plurality of targets.

The processing means may be arranged to perform a method according to the first aspect of the invention.

The location determining means may comprise means for receiving radiation from the object. The location determining means may comprise one or more imaging devices.

Location data indicative of the location of the object at each instant in time may be stored in the memory means.

The system may comprise one or more accelerometers for outputting acceleration data. Advantageously the acceleration data may be used in the determination process, for example to improve the determination e.g. by selecting a model.
The system may comprise a display means for displaying a graphical user interface (GUI) thereon, wherein the plurality of targets are GUI items.

The processing means may be arranged to receive environmental data from one or more sensing means; optionally the sensing means may comprise means for determining a state of the vehicle and/or imaging devices.

According to an aspect of the invention there is provided a vehicle comprising a processing device arranged, in use, to perform a method according to a first aspect of the invention or comprising a system according to the second aspect of the invention.

According to an aspect of the present invention there is provided a method of determining an intended target of an object, comprising determining a location of the object at a plurality of time intervals; determining a probability associated with a target of said target being an intended target.

The probability may be determined based upon a model and the location of the object at the plurality of time intervals.

According to an aspect of the present invention there is provided an apparatus comprising a processing device arranged, in use, to determine an intended target of an object, wherein the processing device is arranged to determine a location of the object at a plurality of time intervals; and to determine a probability associated with a target of said target being an intended target.

Within the scope of this application it is expressly intended that the various aspects, embodiments, examples and alternatives set out in the preceding paragraphs, in the claims and/or in the following description and drawings, and in particular the individual features thereof, may be taken independently or in any combination. That is, all embodiments and/or features of any embodiment can be combined in any way and/or combination, unless such features are incompatible. The applicant reserves the right to change any originally filed claim or file any new claim accordingly, including the right to amend any originally filed claim to depend from and/or
incorporate any feature of any other claim although not originally claimed in that manner.

BRIEF DESCRIPTION OF THE DRAWINGS

Embodiments of the invention will now be described by way of example only, with reference to the accompanying figures, in which:

Figure 1 shows an illustration of fingertip trajectories during pointing gestures;

Figure 2 shows an illustration of a system according to an embodiment of the invention;

Figure 3 illustrates a solid angle for a target;

Figure 4 shows an illustration of a method according to an embodiment of the invention;

Figure 5 shows an illustration of performance of various embodiments of the invention; and

Figure 6 shows a further illustration of performance of various embodiments of the invention;

Figure 7 shows a still further illustration of performance of various embodiments of the invention;

Figure 8 shows a vehicle according to an embodiment of the invention; and

Figure 9 is an illustration of a perturbed trajectory and a filtered trajectory of an object according to an embodiment of the invention.

DETAILED DESCRIPTION

Embodiments of the present invention relate to methods and apparatus for determining an intended target of an object. The object may be a pointing object,
such as a stylus or finger, although it will be realised that this is not limiting. Embodiments of the invention will be explained, by way of example, with reference to fingertip pointing gestures which are performed in vehicles. It will be realised, however, that the pointing object may be an object other than a finger, such an elongate object e.g. a stylus. Furthermore embodiments of the invention are not limited to use within a vehicle and may be used, for example, to determine the intended destination of a pointing object upon a computing device such as a tablet computer or smartphone, for example. Furthermore, embodiments of the invention will be explained with reference to determining the intended destination of the pointing object upon a display device. In particular determining one or a plurality of graphical objects displayed on the display device which is the intended target, or which have a likelihood of being the intended target. It will be realised that embodiments of the invention are not limited the intended target being displayed on a surface of the display device. The display device may be a device for projecting an image onto a surface, such as an interior surface of a vehicle, and detecting the intended target, which may be a graphical object displayed on the surface. For example the surface may be a dashboard or interior portion of the vehicle, although it will be realised that other surfaces may be envisaged. The intended target may also be one of a plurality of physical buttons or other controls, for example. The image may comprise a 3D heliograph and/or a stereoscopic image in some embodiments.

In the description that follows reference will be made to the attached unpublished (draft) papers by the present inventors which form part of the description of the present application. The papers are “Interactive Display in Vehicles: Improving Usability with a Pointing Gesture Tracker and Bayesian Intent Predictors”, “Bayesian Target Prediction From Partial Finger Tracks: Aiding Interactive Displays in Vehicles” and “Filtering Perturbed In-Vehicle Pointing Gesture Trajectories: Improving the Reliability of Intent Inference”. Aspects of embodiments of the present invention will be explained with reference to these papers. Further details of embodiments of the invention may also be provided by the attached papers and such details are included within the present description.

Referring to Figure 1 there is illustrated a fingertip trajectory in three-dimensions (3D) for three separate pointing tasks to select one of a plurality of graphical items displayed on a display device within a vehicle. A location of the fingertip is determined at each of a plurality of time intervals $t_i$ from $t_1$ to $t_k$. At each time interval
a location of the fingertip is determined in 3D as a location vector $\mathbf{m}_n = [x_{tn}, y_{tn}, z_{tn}]^T$. The vector $\mathbf{m}_n$ is used to represent a recorded pointing object location, e.g. of the finger, which may include noise and/or perturbations.

In some embodiments $\mathbf{m}_n$ may be determined with reference to an origin of a sensor arranged to detect the fingertip location, although in other embodiments $\mathbf{m}_n$ may be determined with reference to another location, such as a location within the vehicle, for example a location about the display device. Furthermore, in some embodiments, the vector $\mathbf{m}_n$ may comprise other sensor data such as that output by one or more accelerometers, gyroscopes etc. In other words, the vector $\mathbf{m}_n$ may represent information additional to the position of the object.

Figure 1(a) illustrates the fingertip trajectory 150 (only one of which is numbered for clarity) for three separate pointing tasks to select different graphical items or buttons which are represented as circles 110 (only one of which is numbered for clarity) displayed on a display device 100 in a stationary vehicle. As can be appreciated, even within a stationary vehicle, the trajectories are irregular. Figure 1(b) illustrates trajectories 160 (again only one of which is numbered) for three separate pointing tasks to select different displayed graphical items whilst the vehicle is moving at varying speeds over an uneven road. As can be appreciated the trajectories experience significant perturbations. Other perturbations may arise from, for example, a user walking whilst holding a computing device and attempting a pointing gesture.

Figure 2 illustrates a system 200 according to an embodiment of the present invention. The system 200 is a system for determining an intended target of a pointing object. The system 200 comprises a means 210 for determining a location of the pointing object, a processing means 220 for determining the intended target of the pointing object and a display means 230 for displaying at least one possible target of the pointing object although, as noted above, in other embodiments the possible targets of the pointing object may be a physical object such as a button or other control and thus the display is optional. The processing means 220 may determine whether a target of the pointing object is intended, or has been accidentally targeted. For example whether a graphical item or button was intended to be touched by the user or is touched accidentally such as due to movement of the vehicle. Accordingly the input may be discarded if the processing means 220
determines the target to be unintended. Responsive to the processing means determining the intended target, in some embodiments the display means 230 may be caused to act responsive to the determination to aid a selection process, such as by highlighting the intended target, or one or more possible targets, or to enlarge a portion of information displayed on the display means 230.

In some embodiments the system 200 may include or receive data from one or more additional sensors, such as one or more accelerometers, sensors monitoring a suspension of the vehicle, one or more cameras, such as forward facing to face the road to enable road condition classification, etc. The one or more sensors may help establish an operating environment of the system 200. For example, an accelerometer/camera may be used establish that a lot of vibrations are being or are about-to-be experienced. The one or more accelerometers may enable the system to adapt to prevailing conditions, such as by selecting an appropriate model, as will be explained.

The means 210 for determining the location of the object is a location sensing device 210. The location sensing device may determine the location of the object based on data from one or more devices responsive to received radiation. The radiation may be emitted from one or more devices forming part of the system 200, such as sound waves or electromagnetic radiation. The location sensing device may, in one embodiment, be an accelerometer associated with the object being tracked. The location sensing device may comprise one or more imaging devices for outputting image data relating to the object. The one or more imaging devices may be one or more cameras arranged to output image data including image data corresponding to the object such that the location of the object may be determined therefrom. The location sensing device may be a commercially available device such as a Microsoft Kinect (RTM) or a Leap Motion (RTM) Controller available from Leap Motion, Inc. It will be realised that other devices may be used.

The location sensing device 210 may be arranged to output data from which the location of the object may be determined by the processing means 220 or the location sensing device 210 may output location data indicative of the location of the object. In one embodiment the location sensing device 210 is arranged to output location data at a time instant \( t_k \) of the form \( \mathbf{m}_k \triangleq [\hat{x}_{k|k}, \hat{y}_{k|k}, \hat{z}_{k|k}]^T \) indicative of the location of the object. A value of \( \mathbf{m}_k \), which may be in \( \text{mm} \), may specify the location
of the object with reference to a predetermined datum. The datum may be relative to the location sensing device 210 or may be relative to another datum such as a point about the display device 230.

The location sensing device or the processing means 220 may be arranged to extract or identify the object by performing data association, such as when the location sensing device 210 temporarily loses track of the object. For example, several objects may be detected within a field of vision of the location sensing device 210, such as a pointing hand with several possible fingers, steering wheel, rear viewing mirror, etc. Extracting and/or identifying the desired object such as a pointing finger or other object may be performed as a preliminary step.

The display means 230 is a display device for displaying one or more selectable items which may form part of a graphical user interface (GUI). The display device may be a touch-sensitive screen for outputting visual images comprising the one or more selectable items which may form part of the GUI. The display device 230, in response to a user touching a surface of the screen, may output data indicative of a touched location or may output data indicative of the selected item. In another embodiment the display device 230 may comprise a projection device arranged to project an image onto a surface, such as an interior surface of the vehicle, where the image comprises a selectable object displayed on the surface. For example the surface may be a dashboard or interior portion of the vehicle, although it will be realised that other surfaces may be envisaged.

The processing means 220 may be a processing device comprising one or more processors and memory accessible to the processing device. The memory may store computer software arranged, when executed by the processing device, to perform a method according to an embodiment of the invention. The memory may also, in use, store data indicative of the location of the object at one or more instants in time.

The processing means 220 may comprise a trajectory module 221 for determining the trajectory of the object. It will be realised that the term trajectory may be understood to mean the location of the object at a plurality of instants in time. The trajectory module 221 is arranged to determine a likelihood of one or more possible targets being the intended target of the object.
In particular, the trajectory module 221 may determine, at an instant in time \( t_k \), the probability of a selectable item \( B_i \) being the intended target as \( P(B_i|m_{1:k}) \) where \( m_{1:k} = [b_{x,i} b_{y,i} b_{z,i}]^T \) denotes coordinates of a centre of an \( P \) selectable icon \( B_i \) and \( m_{1:k} = \{m_1, m_2, ..., m_k\} \) comprises all available coordinates of the object at consecutive discrete times \( \{t_1, t_2, ..., t_k\} \). The trajectory module 221 may determine, in some embodiments, \( c_{1:k} = \{c_1, c_2, ..., c_k\} \) as a processed location of the object such as after a pre-processing operation has been performed to, for example, smooth the trajectory of the object. The pre-processing may remove one or more of noise, unintentional movements, vibrations, jumps etc. from the location data \( m_{1:k} \) to produce \( c_{1:k} \). Unintentional movements are, for example, those illustrated in Figure 1(b). It will be appreciated that in the following \( m_{1:k} \) may be replaced with \( c_{1:k} \).

In some embodiments the trajectory module 221 may determine a probability for each of a plurality \( N \) of items \( B = \{B_i; i = 1, 2, ..., N\} \) where \( B \) is a set of items such as selectable GUI items as \( P(B_i|m_{1:k}) \).

A filtering operation may be performed to reduce erratic or unintentional movements of the object. Such movements may be due to road or driving conditions e.g. the road being uneven or the vehicle being driven enthusiastically, such as in a sporting manner. Such movements may also be due to a user walking or moving.

The filtering operation may be a Monte Carlo filtering operation such as Sequential Monte Carlo (SMC). The filtering is performed before an intent inference process, as will be described. The output of the filtering operation at the time instant \( t_n \) is indicative of a true location of the pointing object denoted by \( c_n = [x_{t_n}, y_{t_n}, z_{t_n}]^T \), thus after removing unintentional movements or undesired noise.

For mild perturbations, the filtering operation may be based on linear state space model of the object’s movements. The model may lead to a linear statistical filtering operation, e.g. Linear Kalman filter. More erratic unintentional pointing object movements, e.g. significant jumps or jolts, may be modelled as jumps that may lead to non-linear implementations, e.g. Monte Carlo filtering such as Sequential Monte Carlo (SMC) or Markov Chain Monte Carlo (MCMC) or any other numerical approach.
The probability \( P(B_i|m_{1:k}) \) or \( P(B_i|c_{1:k}) \) of an item being the intended target is determined according to a model and the trajectory of the object. The model may be a linear or a non-linear model. The model may, in some embodiments, model unintended movements such as jumps or jolts due to perturbations i.e. movement such as arising from vehicle movement.

The model may be one of a Nearest Neighbour (NN), Bearing Angle (BA), Heading and Solid Angle (HSA), Linear Destination Reversion (LDR) such as the Mean Reverting Diffusion (MRD) as well as Equilibrium Reverting Velocity (ERV), and Nonlinear Destination Reversion (NLDR). In addition to the information below, further information associated with these models according to embodiments of the invention is provided in the accompanying draft papers.

The intent inference module 222 is arranged to determine an intended target of the object. The intended target is determined using a Bayesian approach. The intent inference module 222 may be arranged to determine the intended target from the plurality \( N \) of targets based on the likelihood associated with each of the plurality of targets \( P(B_i|m_{1:k}) \). This may be equivalent to calculating the Maximum a Posteriori (MAP) via:

\[
\hat{B}(t_k) = \arg \max_{B \in B} P(B_i|m_{1:k})
\]

for the set of \( N \) nominal targets where \( \hat{B}(t_k) \) is the predicted destination and

\[
P(B_i|m_{1:k}) \propto P(m_{1:k}|B_i) P(B_i) \]

according to Bayes' rule \( P(B_i|m_{1:k}) \propto P(m_{1:k}|B_i) \)

The following sections provide a discussion of a plurality of models which may be used by the trajectory module 221.

Nearest Neighbour (NN) Model

In the NN model the likelihood \( P \) is assigned to each item based on a distance to the current position of the object at an instant in time \( t_k \).
This approach chooses the item such as the interface selectable icon that is closest to the current position of the object such as the pointing finger, i.e. $B_i \in B$ with the smallest Euclidean distance $d_{k,i} = \|c_k - b_i\|_2$, $i = 1, 2, ..., N$. In a probabilistic framework, this can be expressed as

$$P(c_k|B_i) = \mathcal{N}(c_k|b_i, \sigma^2_{\alpha\beta})$$  \hfill (5)

The object location $c_k$ has a multivariate normal distribution with a mean equal to that of the possible destination and a fixed covariance $\sigma^2_{\alpha\beta}$. The latter is a design parameter. Assuming that the logged finger positions at various time instants are independent, the sought $P(c_{1:k}|B_i)$ reduces to $P(c_{1:k}|B_i) = \prod_{n=1}^{k} P(c_n|B_i)$.

Bearing Angle (BA) Model

The BA model is based on an assumption that the object moves directly toward the intended destination. The BA model may use the current position of the object at an instant in time $t_k$ and a previous position of the object, which may be $t_{k-1}$. The bearing angle between the positions of the object and the item may be used to calculate the probability.

This model is based on the premise that the pointing finger is heading directly towards the intended destination, i.e. the cumulative angle between the finger positions and the target is minimal. For every two consecutive measurements, the bearing angle with respect to the destination can be assumed to be a random variable with zero mean and fixed variance as per

$$P(c_k|c_{k-1},B_i) = \mathcal{N}(\theta_{i,k}|0, \sigma^2_{\alpha\beta})$$  \hfill (6)

where $\theta_{i,k} = \angle(v_k, b_i)$ for $B_i$, $v_k = c_k - c_{k-1}$ and $\sigma^2_{\alpha\beta}$ is a design parameter. We can write

$$P(c_{1:k}|B_i) = P(c_1|B_i) \prod_{n=2}^{k} P(c_n|c_{n-1},B_i)$$  \hfill (7)

This algorithm can be considered to represent the best outcome of the linear-regression-extrapolation techniques; e.g. assuming that the distance to the intended
destination $d_M$ is accurately estimated. According to (6) and (7), BA forms a wedge-shaped confidence interval whose width is set by $\sigma_{BA}^2$. Any selectable icon that falls within this region is assigned a high probability.

Heading and Solid Angle (HSA) Model

The HSA model is based upon a distance of the object from the item at an instant in time $t_k$ and a solid angle of the item. The HSA model may use the current position of the object at an instant in time $t_k$ and a previous position of the object, which may be $t_{k-1}$.

In the HSA model an object $B_i$ has a smaller solid angle if the observer is far from its location compared with that if the observer is nearby as demonstrated in Fig. 3. Solid angle (in steradians) of a sphere located at distance $d_{i,k}$ is approximated by

$$\Omega_{i,k} = \frac{A \cos(\alpha_k)}{\pi d_{i,k}^2 + \alpha}$$

(8)

where $A$ is the area of the target object. Targets of the arbitrary shapes can be closely approximated by a number of spheres. Parameter $\alpha_k$, which is the exposure angle, is irrelevant to the prediction problem and $\alpha_k = 0$ is assumed. The direction of travel is specified by the measured velocity vector $v_k$ at $t_k$ and the HSA likelihood probability for two consecutive pointing positions can be obtained via

$$P(c_k|c_{k-1}, B_i) = \mathcal{N}(\theta_{i,k}|0, \kappa\Omega_{i,k}).$$

(9)

Similar to the BA model, the divergence of the bearing from the location of $B_i$ is defined by $\theta_{i,k} = z(v_k, b_i)$, $\kappa$ which is a design parameter. If the pointing finger is in close proximity to a possible target bigger $\theta_{i,k}$ values are tolerated due to the resultant $\Omega_{i,k}$. The HSA model can be viewed as a combined BA and NN model. The probability $P(c_{i:k}|B_i)$ can be calculated similar to (7).

It is noted that a distribution other than Gaussian with the relative moments, for example learnt from the collected pointing trajectories, can be applied in the NN, BA and HSA prediction models.
Linear Destination Reverting (LDR) Model

In this approach, the movement of a pointing object is modelled as a function of the intended destination. The characteristics of the pointing movements captured by the adopted model are denoted by a state $s_i$ at time $t$. They can include the pointing object location, multidimensional velocity, multidimensional acceleration, etc. An underlying premise is that the pointing object reverts to the intended destination at a rate that can be specified in the model. A Markov process is then defined where the current pointing movement characteristics is a linear function of the one or more previous moves and the destination. Thus, each of the $N$ possible destinations in a set $B = \{B_i; i = 1, 2, \ldots, N\}$ is associated with a model. The model that matches the characteristics of the pointing object pointing trajectory in the current pointing task is assigned high probability and vice versa. Below we describe two possible LDR models.

Mean Reverting Diffusion (MRD)

The MRD models the object movements as a process that reverts to a particular average value, for example a possible destination. It may only considers the location characteristic of the pointing movement and therefore $s_k = c_k$. It assumes that the current pointing object location should be at the destination that exerts an attraction force to bring the pointing object to its location. In a continuous-time, the pointing object movement is modelled as a multivariate Ornstein-Uhlenbeck process with a mean-reverting term. For the $N$ possible destination, it is described by

$$ds_i(t) = \Lambda(b_i - s_i)dt + \sigma dw_i, \quad i = 1, 2, \ldots, N. \quad (10)$$

The square matrix $\Lambda$ sets the mean reversion rate that steers the evolution of the process, $b_i$ is location of the $i^{th}$ possible destination, $\sigma$ is a square matrix that drives the process dispersion and $w_i$ is a Wiener process. Upon integration of (10) and discretising the outcome, we have:

$$s_{i,k} = e^{-\Lambda k} s_{i,k-1} + [I_3 - e^{-\Lambda k}] b_i + \nu_k, \quad i = 1, 2, \ldots, N \quad (11)$$
where $s_{i,k}$ and $s_{i,k-1}$ are the state vectors with respect to $B_i$ at the time instants $t_k$ and $t_{k-1}$ respectively. The time step is denoted by $\tau_k = t_k - t_{k-1}$ and $v_k \sim N(0, \sigma^2)$ is an additive Gaussian noise.

Equilibrium Reverting Velocity (ERV)

Each of the nominal destinations is assumed to have a gravitational field with strength inversely proportional to distance away from its centre $b_i$. The speed of travel of the object towards the destination location $b_i$ is expected to the highest when the object is far from $b_i$ and vice versa. The movements of the object are modelled with respect to the $i^{th}$ destination as

$$ds_{i,t} = \Lambda (\overline{\mu}_i - s_i)dt + \sigma d\overline{\omega}_i \tag{12}$$

where $s_i = [x_i, \dot{x}_i, y_i, \dot{y}_i, z_i, \dot{z}_i]^T$ such that $\dot{x}_i$, $\dot{y}_i$ and $\dot{z}_i$ are the velocities along the $x$, $y$ and $z$ axes, respectively. Whereas, $\Lambda = diag(\Lambda_x, \Lambda_y, \Lambda_z)$, $\Lambda_x = \begin{bmatrix} 0 & -1 \\ \eta_x & \rho_x \end{bmatrix}$, $\Lambda_y = \begin{bmatrix} 0 & -1 \\ \eta_y & \rho_y \end{bmatrix}$, $\Lambda_z = \begin{bmatrix} 0 & -1 \\ \eta_z & \rho_z \end{bmatrix}$, $\overline{\mu}_i = [b_{x,i}, 0, b_{y,i}, 0, b_{z,i}, 0]^T$ encompassing the coordinates of $B_i$ and $\overline{\omega}_i$ is a Wiener process. Each of $\eta_x$, $\eta_y$ and $\eta_z$ dictates the restoration force along their corresponding axis; $\rho_x$, $\rho_y$ and $\rho_z$ represents a damping factor to smooth the velocity transitions. After integrating (12), we can represent the discretised resultant by

$$s_{i,k} = \tilde{F}_k s_{i,k-1} + \kappa_{i,k} + \tilde{w}_k \tag{13}$$

$$\tilde{F}_k = diag(e^{-\Lambda_x \tau_k}, e^{-\Lambda_y \tau_k}, e^{-\Lambda_z \tau_k}), \kappa = \begin{bmatrix} (1 - e^{-\Lambda_x \tau_k})\overline{\mu}_{x,i} \\ (1 - e^{-\Lambda_y \tau_k})\overline{\mu}_{y,i} \\ (1 - e^{-\Lambda_z \tau_k})\overline{\mu}_{z,i} \end{bmatrix}$$

Given the Gaussian and linear nature of the LDR models, for example (11) and (13), a linear optimal recursive filter can be used to determine the sought $\{P(m_{i,k} | B_i); i = 1,2,\ldots,N\}$ assuming linearly collected measurements $m_k = H_k s_k + n_k$ such that $n_k$ is multivariate Additive White Gaussian Noise. For a destination $B_i$, probability $P(m_{1:k} | B_i)$ can be sequentially calculated since according to the chain rule the following applies $P(m_{i,k} | B_i) = P(m_{i,k} | m_{1:k-1}, B_i), \ldots, P(m_i | m_1, B_i) \times P(m_1 | B_i)$. This implies that at time $t_k$, only the predictive probability $P(m_k | m_{1:k-1}, B_i)$ is required to
determine $P(m_{1,k} | B_i)$ for the $i^{th}$ nominal destination. The pursued $P(m_k | m_{1:k-1}, B_i)$ can be obtained from a Linear Kalman Filter (LKF) whose purpose here is not to track the object, but to produce the predictive probability. As a result, the predictor compromises $N$ Kalman filters each dedicated to a particular nominal suspected destination.

Linear destination reverting models, other than the MRD and ERV, that include more movement characteristics such as acceleration or jerks can be applied. Their implementation is similar to the MRD and ERV models via a bank of statistical filters.

Nonlinear Destination Reverting (NLDR) Model

In this approach, the movements of an object is assumed to include the destination, the characteristics of the pointing movements and nonlinear phenomena such as jumps or jolts representing perturbations in the pointing trajectory due to external factors. An example is carrying out a pointing task in a vehicle moving over harsh terrain as in Figure 1b. An example of a perturbations process is the jump process $p_i$ which represent factors that knocks the pointing object off its planned trajectory. For example, $dp_i = \sigma_p dW_{2,i} + \sigma_j dJ_i$ where the jump process is $J_i = \sum_i P_i$, $P_i \sim \mathcal{N}(0, 1)$ and $l$ is the number of jumps/jolts. The jumps effect allows occasional large impulsive shocks to the pointing object location, velocity, acceleration, permitting the modelling of sharp jolts or sudden movements. Other nonlinear models that capture the characteristics of the present perturbations characteristics may be considered. The model state for each nominal destination $s_{i,t}$ in the NLDR incorporates the pointing object position $c_t = [x_t, y_t, z_t]^T$, other characteristics of $c_t$ (for example velocity $\dot{c}_t$, or acceleration $\ddot{c}_t$, etc.), perturbations $p_t$, other characteristics of $p_t$ and the destination $B_i$.

Similar to LDR model the underlying premise is that the pointing object reverts to the intended destination at a rate that can be specified in the model. A Markov process is then defined where the current pointing movement characteristics is a linear function of the one or more previous moves, the present nonlinear perturbations and the destination. Thus, each of the $N$ possible destinations in the set $ \mathcal{B} = \{B_i; i = 1, 2, \ldots, N\}$ is associated with a model. The model that matches the characteristics of the pointing object pointing trajectory in the current pointing task is assigned high probability and vice versa. Accordingly, a bank of $N$ statistical filters are applied to
sequentially obtain the sought \( \{P(m_{i:k} \mid B_i), i = 1, 2, \ldots, N\} \). Approaches such as sequential Monte Carlo methods or other numerical techniques can be utilised to attain the pursued \( P(m_{1:k} \mid B_i) \) given the nonlinear nature of the state evolution equation once the nonlinear perturbations are included. Minimising the computational complexity of the nonlinear filtering approaches can be achieved by assuming that the perturbations such as jumps or jolts are identical in the bank of \( N \) statistical filters. Hence, they need to be tracked or identified only once.

If the observations model for the LDR or NLDR is not linear or present noise is non-Gaussian, for example \( m_k = f_k(s_k) + n_k \) where \( f_k(.) \) is a nonlinear function, alternative statistical filtering approaches such as sequential Monte Carlo methods or other numerical techniques may be utilised to attain the pursued \( P(m_{1:k} \mid B_i) \).

While the processing means 220 produces the probability of each target being the destination, it might be desirable to sequentially obtain in real-time the underlying unperturbed pointing object trajectory or its characteristics represented by \( s_k \), thus after removing unintentional movements or the present perturbations. This can either be achieved by combining the results of the \( N \) statistical filters used for intentionality prediction or to perform the smoothing operation as a pre-processing stage that precedes calculating \( \{P(m_{1:k} \mid B_i), i = 1, 2, \ldots, N\} \). In the former, it is equivalent to calculating the posterior distribution of the state \( s_k \) at the time instant \( t_k \): \( s_k \) incorporates the pointing object location \( c_k \). The distribution is given by \( P(s_k \mid m_{1:k}) = \sum_{i=1}^{N} P(s_k \mid B_i, m_{1:k}) P(B_i \mid m_{1:k}) \) where \( P(B_i \mid m_{1:k}) = \frac{P(m_{1:k} \mid B_i)P(B_i)}{\sum_{i=1}^{N} P(m_{1:k} \mid B_i)P(B_i)} \) such that \( P(s_k \mid m_{1:k}, B_i) \) is produced by the sequential state update of the statistical filter and \( P(B_i \mid m_{1:k}) \) for \( i = 1, 2, \ldots, N \) is a determined constant. The summation in \( P(s_k \mid m_{1:k}) \) results in a mixed Gaussian model with the minimum mean squared error or a maximum a posteriori estimators of \( s_k \) being the mean and mode of the resultant distribution, respectively.

Removing the perturbations prior to calculating \( \{P(m_{1:k} \mid B_i), i = 1, 2, \ldots, N\} \) to establish the intended destination or destinations entails modelling the pointing process as the sum of the intentional pointing object movements plus unintentional perturbations or noise. In this case, the observed pointing object location using a pointing object tracker module 210 can be modelled as

\[
m_k = s_k + p_k + \epsilon_k
\]
where the unintentional perturbations-related movements and their characteristics are captured in \( p_k \) and the measurement noise is denoted by \( \varepsilon_k \). Various perturbation models can be used including the jump diffusion model. The true pointing movement and/or its characteristics can be modelled using a linear model, thus \( s_k = F_k s_{k-1} + v_k \) where \( s_k \) incorporates the location of the pointing object, velocity, acceleration, etc. Whereas, \( F_k \) is the state transition matrix and \( v_k \) is the present noise. Nearly constant velocity or acceleration models can be used to model the pointing movement in this case, which is independent of the destination. Statistical filtering approaches can be applied to extract \( s_k \) from \( m_k \) by removing or suppressing the unintentional perturbations-related movements. Such techniques include Kalman filtering in case of linear state and perturbations models. Various adapted version of Kalman filtering, sequential Monte Carlo methods or other numerical techniques can be utilised for nonlinear state or observation models.

Figure 9 illustrates a trajectory 910 of an object which exhibits perturbations due, for example, to movement of a vehicle in which the object is moving. A filtered trajectory 920 of the object is also shown which exhibits a more direct course toward the intended target.

It has been observed by the present inventors that only a weak correlation exists between acceleration determined from data output by the location sensing device 210 and that measured by an Inertia measurement unit (IMU) or accelerometer. Thus, whilst use of the IMU data to compensate for noise in the location measurements may not be effective, the IMU data may be used for modifying applied pre-processing and/or the model.

The processing means 220 may comprise an intent inference module 222 for determining the intended target \( \hat{B}(t_k) \) of the object at time instant \( t_k \).

Determining the intended destination, or a number of possible destinations, or the area of the possible destinations at the time instant \( t_k \) relies on the calculated probabilities \( P(B_i|m_{1:k}) \) for \( i = 1, 2, ..., N \). The decision may be based on a cost function \( C(B_i, B^*) \) that ranges from 0 to 1. It penalises an incorrect decision where \( B_i \) is the predicted destination and \( B^* \) is the true intended target in the considered pointing task. For example predicting the wrong destination may impose a maximum
cost of 1. Therefore, the objective is to minimise the average of the cost function in a
given pointing task given the partially observed pointing trajectory \( m_{1:k} \) according to

\[
\hat{B}(t_k) = \arg\min_{B_i \in \mathcal{B}} \mathbb{E}[c(B_i, B^*)|m_{1:k}]
\]

where \( \mathbb{E}[.] \) is the mean. Assume the hard-decision criterion where \( c(B_i, B^*) = 0 \) if
\( B_i = B^* \) and \( c(B_i, B^*) = 1 \) otherwise leads to selecting one target out of the set
\( \{B_i: i = 1, 2, \ldots, N\} \). In this case, it is equivalent to determining the MAP destination
estimate. Other cost function formulations that reflect the desired level of predication

certainty may be used and subsequently a group of selectable targets may be
selected in lieu of one as with the MAP case.

The Bayesian approach relies on a belief-based inference followed by a classifier.
Since the aim is to utilise the available pointing trajectory to determine the
destination, a uniform prior may in some embodiments be assumed on all items, for
example \( P(B_i) = 1/N \) for \( i = 1, 2, \ldots, N \). In this case, the classification problem

corresponds to the maximum likelihood estimation and the solution relies solely on
establishing \( P(m_{1:k}|B_i) \) for \( i = 1, 2, \ldots, N \). However, in other embodiments a non-
uniform prior may be used for the items. For example information concerning
previous selections from a GUI may be used as the prior such that the likelihood of
the intended destination is influenced by a history of user selections. It will be
realised that the prior may alternatively or additionally be based on other information.

In some embodiments only a last \( L \) logged true object positions i.e. \( c_{k-L:k} \triangleq \{c_{k-L}, c_{k-L+1}, \ldots, c_k\} \) and \( k - L > 0 \) may be used to determine \( \hat{B}(t_k) \). In these
embodiments a sliding time window is applied to the trajectory data and a width of
the window may be chosen appropriately.

Figure 4 illustrates a method 400 according to an embodiment of the invention. The
method 400 may be performed by the system 200 described with reference to Figure
2.

In step 410 a location of the object at an instant in time is determined. The location
of the object may be determined by the location sensing device 210 receiving
radiation, such as light or sound, reflected from the object and, from the received
radiation, determining at the time instant \( t_k \) location data as \( m_k \triangleq [x_{tk}, y_{tk}, z_{tk}]^T \).
indicative of the location of the object. The location data may be stored in a memory to form data indicative of a trajectory of the object over a period of time.

In step 420 a likelihood of one or more items being the intended target of the object is determined. The likelihood $P$ may be determined as $P(B_i|m_{t,k})$ as explained above. Step 420 may be performed by the trajectory module 221, as previously explained. In some embodiments the likelihood for each of a plurality of items as $P(B_i|m_{t,k})$ being the intended destination is determined in step 420. The likelihood for the one or the plurality of items being the intended destination is determined based upon a model and the location of the object determined in step 410.

In step 430 the intended target is determined. The intended target may be determined from the likelihood for each of a plurality of items as $P(B_i|m_{t,k})$. Step 430 may be performed by the intent inference module 222 as discussed above. Step 430 may comprise determining the Maximum a Posteriori (MAP).

In some embodiments the method 400 comprises a step 440 in which an output is determined based on the result of step 430. The output may comprise a selection or operation of the intended target. That is, where the intended target is a user-selectable item on the GUI, the item may be selected as though the user had touched the display device to select the item. Alternatively where the intended target is a button or control the button or control may be activated.

The output may be provided via the display device 230. The output may be a modification of the GUI displayed on the display device responsive to the determination of the intended target in step 430. The output may only occur once the likelihood associated with the intended target reaches a predetermined probability $P$, thereby avoiding the item being selected when the likelihood is relatively low. In some embodiments the output of step 440 may comprise a modification to the appearance of the GUI. For example the intended target may be highlighted on the GUI. The intended target may be highlighted when the likelihood associated with the intended target reaches a predetermined probability $P$. The predetermined probability may be lower than that for selection of the intended target, such that, at a first lower probability the intended target is visually indicated and at a second higher probability the intended target is automatically selected. In another embodiment a group of intended targets may be visually indicated in the GUI when
their associated likelihoods of being the intended target are at least the predetermined probability \( P \).

In step 450 it is determined whether the method is complete. If the method is not complete, then the method returns to step 410. If, however, the method is complete then the method ends. The method 400 may be complete when the likelihood associated with one or more items reaches a predetermined threshold probability. For example the method 400 may end when the likelihood reaches the second probability discussed in relation to step 440 at which the intended target is automatically selected.

Figure 5 illustrates results of an experiment at predicting an intended item on a GUI against a percentage of completed pointing movement i.e. \( 100 \times \frac{t_k}{t_M} \) and averaged over all considered pointing tasks \( (t_M \text{ is the total pointing task completion time}) \). Results using the NN, BA, MRD and ERV models are illustrated. Figure 5 starts after completing 15\% of the pointing trajectory duration prior to which none of the techniques produce meaningful results. To represent the level of average prediction uncertainty, Fig. 6 displays the mean of the uncertainty metric given by

\[
\mathcal{E}(t_k) = -\log_{10}(P(B^*(t_k)|m_{1:k}))
\]

where \( P(B^*(t_k)|m_{1:k}) \) is the calculated probability of the true intended item according to the prediction model at time instant \( t_k \). If the true target is predicted with high certainty, i.e. \( P(B^*(t_k)|m_{1:k}) \to 1 \), the confidence in the prediction will be very high as \( \mathcal{E}(t_k) \to 0 \). It is noted that the level of the predictor’s success in inferring the destination does not necessarily imply high prediction certainty and vice versa. In all the simulations, we do not assume that the predictor knows the proportion of the completed trajectory when making decisions. It can be noticed from Fig. 5 that the proposed Bayesian approach provides the earliest successful predictions of the intended target, especially in the crucial first 15\% to 75\% of the pointing movement duration. This success can be twice or three times that the nearest examined competitor. Both MRD and ERV models exhibit similar behaviour, with MRD prediction quality marginally and temporarily degrading in the 70\%-80\% region. This can be due to a failed prediction in a single experiment. Both of these models provide significant performance improvements compared with other techniques. The NN method tends to make successful predictions only in the final portion of the pointing

22
task since the user's finger is inherently close to the intended item at this stage, i.e. briefly before the selection action. In practice, an early prediction, e.g. in the first 75% of the pointing task duration, is more effective at minimising the user movement/cognitive effort, enabling early pointing facilitation techniques and enhancing the overall user experience. The benefits of successful intent inference in the last 25% of the pointing gesture duration are questionable since the user has already dedicated the necessary effort to execute the selection task. The proposed predictors notably outperform the NN for the majority of the duration of the pointing task (or all in the ERV case). With regards to the prediction uncertainty, Fig. 6 shows that the introduced Bayesian predictions can make correct classification decisions with substantially higher confidence levels compared with other techniques. This advantage over the NN model inevitably diminishes as the pointing finger gets closer to the interface in the last portion of the pointing gesture period, e.g. after completing over 75% of the pointing movement.

Figure 7 provides a similar plot to Figure 5 illustrating prediction based on the NN, BA, HSA and MRD models. Again it can be noticed from Figure 7 that the MRD model provides the earliest successful predictions of the intended destination, especially in the crucial first 85% of the pointing gesture.

It can be appreciated that embodiments of the present invention provide methods and apparatus for determining an intended target of an object, where the object may be a pointing object such as a stylus or finger, although the invention is not limited in this respect. The intended target may be one or more intended targets from a plurality of possible targets. The possible targets may be items in a GUI or physical controls. Advantageously embodiments of the present invention may reduce errors associated with HMI, such as by detecting when a selected target was not the intended target i.e. the user accidentally selected a GUI item due to, for example, vehicle movement. Advantageously embodiments of the invention may also reduce a gesture time by selecting a target before a user is able to physically touch the target. Embodiments of the invention may be useful in vehicles such as land vehicles, as illustrated in Figure 8 which comprises a system according to an embodiment of the invention or a processing device arranged to perform a method according to an embodiment of the invention, but also aircraft and watercraft. Embodiments of the invention may also be useful with computing devices such as
portable computer devices e.g. handheld electronic devices such as smartphones or tablet computing devices.

It will be appreciated that embodiments of the present invention can be realised in the form of hardware, software or a combination of hardware and software. Any such software may be stored in the form of volatile or non-volatile storage such as, for example, a storage device like a ROM, whether erasable or rewritable or not, or in the form of memory such as, for example, RAM, memory chips, device or integrated circuits or on an optically or magnetically readable medium such as, for example, a CD, DVD, magnetic disk or magnetic tape. It will be appreciated that the storage devices and storage media are embodiments of machine-readable storage that are suitable for storing a program or programs that, when executed, implement embodiments of the present invention. Accordingly, embodiments provide a program comprising code for implementing a system or method as claimed in any preceding claim and a machine readable storage storing such a program. Still further, embodiments of the present invention may be conveyed electronically via any medium such as a communication signal carried over a wired or wireless connection and embodiments suitably encompass the same.

All of the features disclosed in this specification (including any accompanying claims, abstract and drawings), and/or all of the steps of any method or process so disclosed, may be combined in any combination, except combinations where at least some of such features and/or steps are mutually exclusive.

Each feature disclosed in this specification (including any accompanying claims, abstract and drawings), may be replaced by alternative features serving the same, equivalent or similar purpose, unless expressly stated otherwise. Thus, unless expressly stated otherwise, each feature disclosed is one example only of a generic series of equivalent or similar features.

The invention is not restricted to the details of any foregoing embodiments. The invention extends to any novel one, or any novel combination, of the features disclosed in this specification (including any accompanying claims, abstract and drawings), or to any novel one, or any novel combination, of the steps of any method or process so disclosed. The claims should not be construed to cover merely the
foregoing embodiments, but also any embodiments which fall within the scope of the claims.

Once the claims have been agreed, please provide a set of US-style claims (i.e. without means-plus-function language or multiple dependencies) in the form of a series of numbered paragraphs here at the end of the specific description, for example;

Aspects of the present invention are set forth in the following numbered paragraphs.

1. A method of determining an intended target of an object, comprising determining a location of the object at a plurality of time intervals; determining a metric associated with each of a plurality of targets, the metric indicative of the respective target being the intended target of the object, wherein the metric is determined based upon a model and the location of the object at the plurality of time intervals; determining, using a Bayesian reasoning process, the intended target from the plurality of targets based on the metric associated with each of the plurality of targets.

2. The method of paragraph 1, comprising determining a trajectory of the object.

3. The method of paragraph 2, wherein the trajectory of the object comprises data indicative of the location of the object at a plurality of time intervals.

4. The method of paragraph 2, comprising filtering the trajectory of the object.

5. The method of paragraph 4, wherein the filtering smooths the trajectory of the object and/or the filtering reduces unintended movements of the object and/or noise from the trajectory.

6. The method of paragraph 1, wherein the model is a Bayesian intentionality prediction model.

7. The method of paragraph 1, wherein the model is a linear model.
8. The method of paragraph 7, wherein the linear model is based on one or more filters; optionally the one or more filter are Kalman filters.

9. The method of paragraph 1, wherein the model is a non-linear model.

10. The method of paragraph 9, wherein the non-linear model incorporates irregular movements of the object.

11. The method of paragraph 9, wherein the non-linear model is based on one or more particle filters.

12. The method of paragraph 1, wherein the model is a nearest neighbour (NN) model.

13. The method of paragraph 12, wherein the NN model determines the metric based upon a distance between the location of the object and each of the targets.

14. The method of paragraph 12, wherein the metric is indicative of a distance between the object and each of the targets.

15. The method of paragraph 1, wherein the model is a bearing angle (BA) model.

16. The method of paragraph 15 when dependent upon paragraph 2 or any paragraph dependent thereon, wherein the metric is indicative of an angle between the trajectory of the object and each of the targets.

17. The method of paragraph 1, wherein the model is a heading solid angle (HSA) model.

18. The method of paragraph 17, wherein the metric is indicative of a solid angle between the object and each of the targets.

19. The method of paragraph 1, wherein the model is a Linear Destination Reversion (LDR) or a Nonlinear Destination Reversion (NLDRe) model.
20. The method of paragraph 19, comprising determining a model for each of the targets.

21. The method of paragraph 20, when dependent upon paragraph 2 or any paragraph dependent thereon, wherein the metric is indicative of the model best matching the trajectory of the object.

22. The method of paragraph 19, when dependent upon paragraph 2 or any paragraph dependent thereon, wherein the NLDR model comprises non-linear perturbations of the trajectory.

23. The method of paragraph 1, wherein the model is a Mean Reverting Diffusion (MRD) model.

24. The method of paragraph 1, wherein the MRD models a location of the object as a process reverting to the intended target.

25. The method of paragraph 1, wherein the model is an Equilibrium Reverting Velocity (ERV) model.

26. The method of paragraph 25, wherein the metric is based upon a speed of travel of the object to the target.

27. The method of any of paragraph 19, comprising determining a state of the object.

28. The method of paragraph 1, comprising receiving one or more items of environmental information.

29. The method of paragraph 28, wherein the environmental information comprises one or more of: information indicative of acceleration, information indicative of a state of the vehicle and/or image data indicative of vehicle surroundings.

30. The method of paragraph 28, wherein the determination of the metric is based, at least in part, on the one or more items of environmental information.
31. The method of paragraph 28, wherein the model is selected based, at least in part, on the one or more items of environmental information.

32. The method of paragraph 1, wherein the determining the intended target is based on a cost function.

33. The method of paragraph 1, wherein the cost function imposes a cost for incorrectly determining the intended target.

34. The method of paragraph 32, wherein the intended target is determined so as to reduce the cost function.

35. The method of paragraph 1, wherein the determining the intended target is based on one or more items of prior information.

36. The method of paragraph 35, wherein the prior information is associated with at least some of the targets.

37. The method of paragraph 1, comprising selecting a plurality of most recent time intervals, wherein the determining the metric associated with each of the plurality of targets is based upon the location of the object at the plurality of most recent time intervals.

38. The method of paragraph 1, wherein the object is a pointing object.

39. The method of paragraph 1, wherein the location of the object is determined in three dimensions.

40. The method of paragraph 1, wherein determining the location of the object comprises tracking the location of the object.

41. The method of paragraph 1, wherein determining the location of the object comprises receiving radiation from the object.
42. The method of paragraph 1, comprising outputting an indication of the intended target.

43. The method of paragraph 42, wherein the indication of the intended target comprises identifying the intended target; optionally the intended target is visually identified.

44. The method of paragraph 42, comprising outputting the indication of the intended target and one or more possible targets.

45. The method of paragraph 1, comprising activating the intended target.

46. The method of paragraph 1, wherein the plurality of targets comprise one or more of graphically displayed items or physical controls.

47. The method of paragraph 1, wherein the location of the object is determined in three-dimensions.

48. A system for determining an intended target of an object, comprising location determining devices for determining a location of the object; a memory device for storing data indicative of the location of the object at one or more instants in time; a processing device arranged to determine a metric associated with each of a plurality of targets of the respective target being the intended target of the object, wherein the metric is determined based upon a model and the location of the object at the plurality of time intervals; determine, using a Bayesian reasoning process, the intended target from the plurality of targets based on the metric associated with each of the plurality of targets.

49. The system of paragraph 48, wherein the processing device is arranged to perform a method as claimed in any of paragraph 2 to 47.

50. The system of paragraph 48, wherein the location determining device comprises devices for receiving radiation from the object.

51. The system of paragraph 48, wherein the location determining device comprises one or more imaging devices.
52. The system of any of paragraph 48, wherein location data indicative of the location of the object at each instant in time is stored in the memory device.

53. The system of any of paragraphs 48, comprising one or more accelerometers for outputting acceleration data.

54. The system of any of paragraph 48, comprising a display device for displaying a graphical user interface (GUI) thereon, wherein the plurality of targets are GUI items.

55. The system of any of paragraph 48, wherein the processing device is arranged to receive environmental data from one or more sensing devices; optionally the sensing device comprise devices for determining a state of the vehicle and/or imaging devices.

56. A vehicle comprising a processing device arranged, in use, to perform a method according to paragraph 1 or comprising the system of paragraph 48.
APPENDIX

Bayesian Target Prediction From Partial Finger Tracks: Aiding Interactive Displays in Vehicles

(a) Vehicle is stationary.

(b) Vehicle is moving at varying speeds and over uneven road.

We formulate the prediction problem within a Bayesian tracking framework by adopting state space models that incorporate the selectable targets. Tracking and intent inference are combined. A Linear Kalman Filter (LKF) based solution is introduced to dynamically obtain the posterior probability, for each of the possible destinations given the available partial pointing finger trajectory. Selectable interface targets that produce predictive trajectories that fit the observed track are assigned high probabilities. Thus, it is a belief-based prediction and a cost function can be used to decide the intended destination. Priors on the selectable targets, if available, can be easily included to improve the intention sensitivity.

II. PROBLEM FORMULATION AND PROPOSED APPROACH

Let \( m_{i,k} \equiv \{ m_{i1}, m_{i2}, ..., m_{ik} \} \) be the available pointing finger tip tracks at certain discrete-times up to \( t_k \). We recall that at time instant \( t_m \), the observation vector produced by the LM controller \( m_{in} \) comprises the pointing finger tip Cartesian coordinates. Define \( B = \{ B_i : i = 1, 2, ..., N \} \) as the set of \( N \) selectable items on the touchscreen interface. No assumptions are made about their distribution and their locations are known where \( b_i = [b_{i,1}, b_{i,2}, b_{i,3}]^T \) denotes the coordinates of the \( i^{th} \) button with respect to the LM sensor position. Let \( t_m \) be the total pointing time taken for the user to reach the touchscreen and make a selection. The objective here is to infer the intended target on the touchscreen interface from \( m_{1:0} \) where \( t_k \leq t_m \). At time \( t_c \), the inference algorithm reached a correct decision, the achieved reduction in the pointing duration is \( t_{c} - t_{m} \).

With the arrival of new observation \( m_n \), the proposed predictor follows two consecutive steps: 1) estimates the likelihood values \( P(t_n) = \{ P(B_i | m_{i:k}), i = 1, 2, ..., N \} \) and 2) makes a decision on the intended target based on a chosen cost function. Thus, the adopted Bayesian approach relies on a belief-based inference followed by a classifier. The likelihood values in \( P(t_n) \), each corresponding to each selectable item \( B_i \), can be expressed by

\[
P(B_i | m_{i:k}) \propto P(B_i)P(m_{i:k} | B_i).
\]

where \( \hat{B}(t_n) \) is the inferred destination, \( C(B_i, B^*) \) is the decision cost of choosing button \( B_i \) as the destination given \( m_{i:k} \) and \( B^* \) is the true intended target. For simplicity, we assume the following hard-decision criterion

\[
C(B_i, B^*) = \begin{cases} 
0, & B_i = B^* \\
1, & B_i \neq B^*.
\end{cases}
\]

Since \( \mathbb{E}[C(B_i, B^*) | m_{i:k}] = \sum_{B_i \in B} C(B_i, B^*) P(B_i | m_{i:k}) \), it can be easily seen that (4) is equivalent to determining the Maximum a Posteriori (MAP) destination estimate

\[
\hat{B}(t_n) = \arg \max_{B_i \in B} P(B_i | m_{i:k}) P(B_i),
\]

which is equivalent to the Maximum Likelihood (ML) when a uniform prior is used. More elaborate cost function formulations can be devised and a group of selectable targets, e.g. \( B(t_n) \), could even be selected in lieu of one. Exploring the costing strategy is outside the scope of this paper and the ML/MAP estimator is henceforth adopted. Additionally, it might be desirable in certain scenarios to utilise only the last \( L \) logged finger positions, i.e. \( m_{-L+1:k} \equiv \{ m_{-L+1}, m_{-L+2}, ..., m_{k} \} \) and \( k - L > 0 \), to determine \( \hat{B}(t_n) \). A sliding time window is
applied to the data and the window width is a design parameter. It is noted that determining (3) and thereby (6) depends solely on \( P(m_{i+k} | B_t) \). Next, two algorithms that yield \( P(m_{i+k} | B_t) \) are introduced and subsequently evaluated.

III. BAYESIAN PREDICTORS

Let \( v_t = [x_t, y_t, z_t] \) be the vector that represents the true location of the pointing finger at \( t \), according to a predefined dynamics model. Two linear process models, namely Mean Reverting Diffusion (MRD) and Equilibrium Reverting Velocity (ERV), are utilized in this section and identified to them are presented in Appendix A. The analytical relationship between the model state vector \( s_t \) and the observation vector \( m_t \) is described by

\[
m_t = \Pi_t s_t + n_t.
\]

where \( s_t \in \mathbb{R}^2 \) is zero mean multivariate Additive White Gaussian Noise (AWGN) with covariance matrix \( \Sigma_t = \mathbb{E}[n_t n_t^T] \). \( \Pi_t \) is the transpose of vector/matrix \( X_t \). For the MRD case, \( s_t = \hat{s}_t = c_0 \) and \( \Pi_t = I_3 \) is a \( 3 \times 3 \) identity matrix. Whereas, for the ERV model we have \( s_t = \hat{s}_t = [x_t, \dot{x}_t, y_t, \dot{y}_t, z_t, \dot{z}_t] \) such that \( \dot{x}_t, \dot{y}_t \) and \( \dot{z}_t \) are the velocities along the \( x, y \) and \( z \) axes respectively. Its observation matrix is

\[
\hat{\Pi}_t = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}.
\]

A. MRD Model

In continuous-time, we model the movements of the user pointing finger as a multivariate Ornstein-Uhlenbeck process with a mean-reverting term. It is described by

\[
d\hat{s}_t = \Lambda \left( \hat{s}_t - \hat{s}_t \right) dt + \sigma d\tilde{w}_t,
\]

such that \( \Lambda \) is a square matrix that sets the mean reversion rate which steers the evolution of the process, \( \hat{s}_t \) is the process mean, \( \sigma \) is a square matrix that drives the process dispersion and \( \tilde{w}_t \) is a Wiener process. The MRD model is an intuitive candidate for the task of predicting the evolution of the pointing finger, as users tend to move relatively fast towards the intended target button with frequent diversions from the shortest trajectory. Hence, \( \hat{s}_t \) for the \( i^i \) button.

By integrating (8) over the time interval \( T = [t, t + \tau] \) and then discretising the outcome (since the finger positions are recorded at discrete-time instants), we obtain

\[
\hat{s}_{i,k} = \tilde{F}_k \hat{s}_{i,k-1} + \tilde{K}_{i,k} + \tilde{w}_k \quad \text{where} \quad \tilde{w}_k \sim \mathcal{N}(0, \tilde{\Sigma}_k),
\]

such that \( \tilde{w}_k = i_k - b_k \) is the time step size. The dynamic noise is given by \( \tilde{w}_k \sim \mathcal{N}(0, \tilde{\Sigma}_k) \). Assuming that the movements of the finger along the \( x, y \) and \( z \) axes are independent, the \( \Lambda \) and \( \sigma \) matrices become diagonal, i.e., \( \Lambda = \text{diag} \{ \lambda_x, \lambda_y, \lambda_z \} \) and \( \sigma = \text{diag} \{ \sigma_x, \sigma_y, \sigma_z \} \). Noting that \( \Lambda \) is positive definite, it follows that the distribution of the conditional state is given by

\[
P(\hat{s}_{i,k} | \hat{s}_{i,k-1}) = \mathcal{N} \left( \hat{s}_{i,k} | \tilde{F}_k \hat{s}_{i,k-1} + \tilde{K}_{i,k}, \tilde{\Sigma}_k \right)
\]

such that

\[
\tilde{F}_k = e^{-\Lambda \tau_k}, \quad \tilde{K}_{i,k} = \left[ I_3 - e^{-\Lambda \tau_k} \right] b_k
\]

and

\[
\tilde{\Sigma}_k = \left[ I_3 - e^{-2\Lambda \tau_k} \right] \sigma^2
\]

It is noted that index \( i \) can be discarded whenever conditioning on \( B_t \) is explicitly stated; i.e., \( P(\hat{s}_{i,k} | \hat{s}_{i,k-1}, B_t) \) replaces \( P(\hat{s}_{i,k} | \hat{s}_{i,k-1} - 1, B_t) \) and \( \hat{s}_{i,k} = c_0 \) refers to the true state representing the position of the pointing finger in 3D at \( t \), regardless of the intended destination.

B. ERV Model

Unlike the MRD model, each of the nominal target buttons here is assumed to have a gravitational field with strength inversely proportional to distance away from its centre \( b_k \). The speed of travel towards the equilibrium point \( b_k \) is expected to the highest when the finger is far from the touchscreen, which is consistent with human behavior during a pointing task as demonstrated in various pointing/facilitation methods [5]-[13]. Thus, we model the movements of the pointing finger with respect to the \( i^i \) button as

\[
d\hat{s}_{i,k} = \Lambda (\hat{s}_t - \hat{s}_t) dt + \sigma \tilde{d}w_t, \quad \tilde{A} = \text{diag} \{ \lambda_x, \lambda_y, \lambda_z \}
\]

\[
\Lambda = \text{diag} \{ \lambda_x, \lambda_y, \lambda_z \}, \quad \sigma = \text{diag} \{ \lambda_x, \lambda_y, \lambda_z \}
\]

\[
\tilde{d}w_t \sim \mathcal{N}(0, \tilde{\Sigma}_k)
\]

\[
P(\hat{s}_{i,k} | \hat{s}_{i,k-1}) = \mathcal{N} \left( \hat{s}_{i,k} | \tilde{F}_k \hat{s}_{i,k-1} + \tilde{K}_{i,k}, \tilde{\Sigma}_k \right)
\]

such that

\[
\tilde{F}_k = e^{-\Lambda \tau_k}, \quad \tilde{K}_{i,k} = \left[ I_3 - e^{-\Lambda \tau_k} \right] b_k
\]

and

\[
\tilde{\Sigma}_k = \left[ I_3 - e^{-2\Lambda \tau_k} \right] \sigma^2
\]

The sub-matrices \( \chi_{i,i}, \chi_{i,j}, \chi_{i,j}, \chi_{i,j} \) and \( \chi_{i,j}, \chi_{i,j}, \chi_{i,j} \) are calculated.

\[
\tilde{d}w_t \sim \mathcal{N}(0, \tilde{\Sigma}_k)
\]

where \( \lambda_x, \lambda_y, \lambda_z \) are the principal axes of \( \sigma \) and \( \tilde{\Sigma}_k \) is a square diagonal matrix with the \( \Sigma \) entries along its diagonal. Matrix entries result in a square block diagonal matrix.
C. Bayesian Sequential Intent Inference

Given the Gaussian and linear nature of (7), (9) and (15), a linear optimal recursive filter can be used to determine the sought \( P(m_{i:k} | B_i) ; i = 1, \ldots, N \) and implement the MAP estimator in (3) and (6). It is optimal in the Minimum Mean Squared Error (MMSE) sense [17]. According to the chain rule, we can write:

\[
P(m_{i:k} | B_i) = P(m_{i:k-1} | m_{i-1:k-1}, B_i), \ldots, P(m_2|m_1, B_1) \times P(m_1|B_1).
\]

This implies that at \( t_k \), we only need to obtain the predictive probability \( P(m_{i:k} | m_{i-1:k-1}, B_i) \) to sequentially determine \( P(m_{i:k} | B_i) \). Hence the proposed MAP estimators follow identical steps to that of the Kalman filter, with the distinct objective of estimating the predicted observations distribution \( P(m_{i:k} | B_i) \) for each of the \( N \) buttons rather than tracking the state \( s_k \).

For a vector \( \mathbf{e}_k \in \mathbb{R}^n \), let \( \mathbf{e}_{i:k-1} = \mathbb{E}[e_k | m_{i:k-1}, B_i] \) and \( \mathbf{P}^{\text{est}}_{i:k-1} = \mathbb{E} \left[ (e_k - \mathbf{e}_{i:k-1}) (e_k - \mathbf{e}_{i:k-1})^T \right] \) where

\[
\mathbb{E}[e_k | m_{i:k-1}, B_i] = \sum_{i=1}^{N} P(e_k | m_{i:k-1}, B_i) \mathbb{E}[e_k | B_i] d\mathbb{E}[e_k],
\]

and

\[
\mathbb{E} \left[ (e_k - \mathbf{e}_{i:k-1}) (e_k - \mathbf{e}_{i:k-1})^T \right] = \sum_{i=1}^{N} P(e_k | m_{i:k-1}, B_i) \mathbb{E} \left[ (e_k - \mathbf{e}_{i:k-1}) (e_k - \mathbf{e}_{i:k-1})^T \right] d\mathbb{E}[e_k].
\]

Similarly, we have: \( \mathbf{e}_{i:k-1} = \mathbb{E}[e_k | m_{i:k-1}, B_i] \) and \( \mathbf{P}^{\text{est}}_{i:k-1} = \mathbb{E} \left[ (e_k - \mathbf{e}_{i:k-1}) (e_k - \mathbf{e}_{i:k-1})^T \right] \). Since \( P(m_{i:k} | m_{i-1:k-1}, B_i) \) is \( \int_{\mathbb{R}^n} P(m_{i:k} | m_{i-1:k-1}, B_i) d\mathbb{E}[m_{i-1:k-1}] \), the present measurement noise is zero mean, the sought predictive density can be easily shown to reduce to

\[
P(m_{i:k} | m_{i-1:k-1}, B_i) = N \left( \mathbf{m}_{i:k-1}, \mathbf{P}^{\text{est}}_{i:k-1} \right)
\]

such that

\[
\mathbf{m}_{i:k-1} = H_k \mathbf{s}_{i:k-1}
\]

and

\[
\mathbf{P}^{\text{est}}_{i:k-1} = H_k \mathbf{P}^{\text{est}}_{i:k-1} H_k^T + N
\]

where \( \mathbf{s}_{i:k-1} \) and \( \mathbf{P}^{\text{est}}_{i-1:k-1} \) are the first and second order moments of the predicted state vector conditioned on all but the current observation at time \( t_k \).

The conditional predictive state probability can be expressed by \( P(s_k | m_{i-1:k-1}, B_i) = \mathcal{N} \left( \mathbf{s}_{i:k-1}, \mathbf{P}^{\text{est}}_{i-1:k-1} \right) \) where \( \mathbf{s}_{i:k-1} = \mathbb{E}[s_k | m_{i-1:k-1}, B_i] \) and \( \mathbf{P}^{\text{est}}_{i-1:k-1} = \mathbb{E} \left[ (s_k - \mathbf{s}_{i:k-1}) (s_k - \mathbf{s}_{i:k-1})^T \right] \). From these densities, it is possible to create an approximate particle filter. In order to determine \( \mathbf{s}_{i:k} \) and \( \mathbf{P}_{i:k} \), it is necessary to establish the proposed likelihoods via (23) to (27) at the next time instant \( t_{k+1} \). For the Kalman filtering update equations, we have,

\[
\mathbf{s}_{i:k} = \mathbf{s}_{i:k-1} + \mathbf{G}_{i:k} (m_{i:k} - \mathbf{m}_{i:k-1})
\]

\[
\mathbf{P}_{i:k} = \mathbf{P}_{i:k-1} - \mathbf{G}_{i:k} \mathbf{P}_{i:k-1} \mathbf{G}_{i:k}^T
\]

such that \( \mathbf{G}_{i:k} = \mathbf{P}_{i:k-1} \mathbf{K}_{i:k} \mathbf{P}_{i:k-1} \mathbf{K}_{i:k}^T \) is the Kalman gain and \( \mathbf{P}_{i:k} = \mathbf{P}_{i:k-1} - \mathbf{K}_{i:k} \mathbf{P}_{i:k-1} \mathbf{K}_{i:k}^T \). Fig. 3 depicts the block diagram of the adopted procedure for sequentially determining \( P(m_{i:k} | m_{i-1:k-1}, B_i) \). The latter is used to calculate \( P(m_{i:k} | B_i) \) via (29) to infer the intended destination. We recall that the generic model identities in (24) to (29) are defined by

- MRD model: \( s_k = [x_k, y_k, z_k]^T \), \( \mathbf{f}_k = \mathbf{f}_k \), \( \mathbf{h}_k = \mathbf{h}_k \).
- ERV model: \( s_k = [x_k, y_k, z_k, \omega_k]^T \), \( \mathbf{f}_k = \mathbf{f}_k \), \( \mathbf{h}_k = \mathbf{h}_k \).

D. State Estimation

Determining the true pointing finger position \( \mathbf{e}_k \) might be beneficial to produce the smoothest track by taking into account the intended state dynamics model. This is equivalent to calculating the posterior distribution which is given by

\[
P(s_k | m_{i:k}) = \sum_{i=1}^{N} P(s_k | m_{i:k}, B_i) P(B_i | m_{i:k})
\]

where it can be easily seen that

\[
P(B_i | m_{i:k}) = \frac{P(m_{i:k} | B_i) P(B_i)}{\sum_{i=1}^{N} P(m_{i:k} | B_i) P(B_i)}
\]

It is noted that \( P(s_k | m_{i:k}, B_i) = \mathcal{N} \left( \mathbf{s}_k, \mathbf{P}_{i:k} \right) \) is produced by the sequential state update step in Fig. 3 and \( P(B_i | m_{i:k}) \) for \( i = 1, 2, \ldots, N \) is a constant. The summation in (30) results in a mixed Gaussian model with the MMSE and MAP estimators of \( s_k \) being the mean and mode of the resultant distribution, respectively.

IV. SIMULATIONS

We evaluate the performance of the introduced Bayesian destination inference for the MRD and ERV models on 12 data sets pertaining to typical pointing tasks conducted by three users in stationary and mobile vehicles. In each of these tasks, the user selects a target on the interactive interface by reaching to a touchscreen mounted to the vehicle dashboard and touches the highlighted button. The layout of the interface selectable items is similar to that shown in Fig. 1 with \( N = 71 \) closely clustered buttons (less than 1.2cm apart) in lieu of 19.

Additionally, the following two benchmark techniques are examined:

- NN: this is an intuitive baseline method that allocates the highest probability to the target closest to the current finger tip position; i.e., \( B_i \in B \) with the smallest distance \( d_{ik} = \| m_{i} - b_i \| \). The likelihood of the \( i \)-th button can be expressed via \( P(m_{i} | B_i) = \mathcal{N} \left( m_{i}, \mathbf{b}_i, \sigma_{xy}^2 \right) \) where \( \sigma_{xy}^2 \) is the covariance of the multivariate normal distribution.
- BA: the hearing angle from two consecutive cursor positions with respect to a target can be assumed to be a random variable with zero mean and fixed variance, thus \( P(m_{i} | m_{i-1}, B_i) = \mathcal{N} \left( \mathbf{b}_i, \sigma_{xy}^2 \right) \) where \( \mathbf{b}_i = \mathbb{Z}(x_{i-1}, b_i), \mathbf{b}_i = \mathbb{Z}(x_{i-1} - m_i - m_i-1, \sigma_{xy}^2) \) is a design parameter. The likelihood values are determined via \( P(m_{i} | m_{i-1}, B_i) = \mathcal{N} \left( m_{i-1}, \mathbf{b}_i, \sigma_{xy}^2 \right) \).
It is noted that BA is based on the notation that as the pointing finger is heading towards the target button, the cumulative angle between the direction of travel and the position of the target is minimal. This algorithm can be reasonably considered to represent the best outcome of the linear-regression-extrapolation techniques, e.g., assuming that $d_{23}$ in (1) is accurately estimated. The width of the time window is a design parameter and the value that yields the highest likelihood probability $P(m_{1:k}|B^t)$ for the given data sets is selected. Other design parameters are similarly attained from a preliminary parameters training stage.

Fig. 3: Block diagram for determining the observation predictive density of the $i^{th}$ button for a given dynamics model.

For each data set, an intent prediction is performed at the arrival of each new LM observation indicating the pointing finger coordinates. We assess the performance of a predictor in terms of its classification success from the MAP estimator in (6), i.e., when or how early in the pointing gesture the predictor assigns the highest probability to the true intended target. This is depicted in Fig. 4 against the percentage of completed pointing movement in terms of time, i.e. $100 \times t_{k}/t_{\text{M}}$, and averaged over all considered pointing tasks. We start after completing 15% of the pointing trajectory duration prior to which none of the techniques produce meaningful results. To represent the level of average prediction uncertainty, Fig. 5 displays the mean of the uncertainty metric given by

$$E(t_k) = -\log_{10}(P(B'(t_k)|m_{1:k}))$$

where $P(B'(t_k)|m_{1:k})$ is the calculated probability of the true intended item according to the prediction model at time instant $t_k$. If the true target is predicted with high certainty, i.e. $P(B'(t_k)|m_{1:k}) = 1$, the confidence in the prediction will be very high as $E(t_k) \to 0$. It is noted that the level of the predictor's success in inferring the destination does not necessarily imply high prediction certainty and vice versa. In all the simulations, we do not assume that the predictor knows the proportion of the completed trajectory when making decisions.

It can be noticed from Fig. 4 that the proposed Bayesian approach provides the earliest successful predictions of the intended target, especially in the crucial first 15% to 75% of the pointing movement duration. This success can be twice or three times that of the nearest examined competitors. Both MRD and ERV models exhibit similar behavior, with MRD prediction quality marginally and temporarily degrading in the 70%-80% region. This can be due to a failed prediction in a single experiment. Both of these models provide significant performance improvements compared with other techniques. The NN method tends to make successful predictions only in the final portion of the pointing task, since the user’s finger is inherently close to the intended item at this stage, i.e., briefly before the selection action. In practice, an early prediction, e.g., in the first 25% of the pointing task duration, is more effective at minimising the user movement/cognitive effort, enabling early pointing facilitation techniques and enhancing the overall user experience. The benefits of successful intent inference in the last 25% of the pointing gesture duration are questionable since the user has already dedicated the necessary effort to execute the selection task.

The proposed predictors notably outperform the NN for the majority of the duration of the pointing task (or all in the ERV case). It was shown that NN provides significantly higher quality predictions compared to the optimal-control algorithm in the last 25% to 50% of the pointing duration. This is a critical drawback, which is not present in the MRD model and is negligible for the MRD.

Fig. 4: Mean percentage of destination successful prediction as a function of the percentage of the pointing gesture time.

Fig. 5: Average log prediction uncertainty $\sum_{k=1}^{L} E(t_k)$ as a function of the pointing movement duration.
With regards to prediction uncertainty, Fig. 5 shows that the introduced Bayesian predictions can make correct classification decisions with substantially higher confidence levels compared with other techniques. This advantage over the NN model inevitably diminishes as the pointing finger gets closer to the interface in the last portion of the pointing gesture period, e.g. after completing over 75% of the pointing movement. In Fig. 6, we depict the proportion of the total pointing gesture (in time) the algorithm correctly predicted the true intended item on the interface. Whilst in Fig. 6a this is displayed for each data set (result for each set is identically coloured for all examined algorithms), Fig. 6b shows the average combined results of all the data sets. It is clear from the figure that the proposed MRD-based and ERV-based predictors consistently deliver higher overall success rates compared with NN and BA models in each experiment and indeed on average; ERV slightly outperforms the MRD on average. For example, MRD and ERV models infer the correct intended destination in approximately 65% of the performed predictions or the pointing gesture duration compared to 48.8% with the nearest neighbour approach. The BA approach performs poorly throughout due to the continuously changing heading angle whose mean is well above zero as shown in Fig. 2. For instance, as the pointing finger becomes closer to the target, the heading angle can have arbitrarily values resulting in the high level of classification errors (see Fig. 4) and uncertainty (see Fig. 5).

The above simulations demonstrate the tangible performance improvements provided by the introduced prediction approach. It is important to note that since interactions with displays are very prevalent in modern vehicle environments, small improvements, in pointing task efficiency, even reducing the pointing time by few milliseconds, will have substantial aggregate benefits on safety and the overall user experience, especially for a driving user. A more complete experimental study that considers a large number of subjects and categorises the road as well as driving conditions is needed to best quantify the gains of intent predictors and choose the form(s) of subsequent pointing facilitation technique(s) based on the user experience. Nonetheless, from the conducted relatively large set of data collection experiments, part of which are shown here, the proposed Bayesian approach consistently outperforms other methods.

V. CONCLUSION

This paper sets a Bayesian framework for belief-based target prediction in 3D environments aimed at improving touchscreen usability in vehicles using state-of-the-art finger tracking devices. The two analysed computationally efficient prediction models can provide significant performance improvements. They can significantly reduce the pointing time duration and achieve superior prediction results. This minimises the time, physical effort and cognition the user is required to dedicate to interact with in-vehicle displays, especially the driver.

APPENDIX A

INTEGRATION AND MOMENTS OF A LINEAR MODEL

Both MRD and ERV models can be represented by: $\mathbf{d}_t = \mathbf{A}(\mathbf{a} - \mathbf{s}_t) dt + \mathbf{w}_t$, where $\mathbf{s} \in \mathbb{R}^{n \times 1}$ is the state in question, $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{a} \in \mathbb{R}^{n \times 1}$, $\mathbf{w}_t \in \mathbb{R}^{n \times 1}$ is the state noise. Let $f(s_t, t) = e^{\mathbf{A}^T s_t}$, then we have:

$$df(s_t, t) = \mathbf{A} \mathbf{A}^T e^{\mathbf{A}^T s_t} dt + \mathbf{w}_t e^{\mathbf{A}^T s_t} dt$$

Integrating (32) in $T = [t_1, t_2]$ including the initial value $e^{\mathbf{A}^T s_{t_1}}$ leads to:

$$e^{\mathbf{A}^T s_{t_2}} = e^{\mathbf{A}^T s_{t_1}} + \int_{t_1}^{t_2} e^{A_s^T} \mathbf{A} dt e^{\mathbf{A}^T s_t}$$

Hence,

$$s_{t_2} = e^{-\mathbf{A}_T s_{t_1}} + \int_{t_1}^{t_2} e^{\mathbf{A}(t-t_1)} \mathbf{A} dt e^{\mathbf{A}^T s_t}$$

such that $\mathbf{A} = t_2 - t_1$. and $\mathbf{I}_n$ is a $n \times n$ identity matrix. Noting that $E[d\mathbf{w}_t] = 0$, it can be easily seen that:

$$E[s_{t_2} | s_{t_1}] = e^{-\mathbf{A}_1 s_{t_1}} + \int_{t_1}^{t_2} e^{\mathbf{A}(t-t_1)} \mathbf{A} dt e^{\mathbf{A}^T s_t}$$

With regards to the covariance expression of (34), using Ito's isometry noting that $\text{Cov}[s_{t_2} | s_{t_1}] = E[[s_{t_2} e^{\mathbf{A}(t-t_1)} \mathbf{A} dt e^{\mathbf{A}^T s_t}]]$, we obtain:

$$\text{Cov}[s_{t_2} | s_{t_1}] = \int_{t_1}^{t_2} e^{\mathbf{A}(t-t_1)} \mathbf{A} \mathbf{A}^T e^{\mathbf{A}^T s_t} dt$$

which can be simplified based on the structure of $\mathbf{A}$ and $\mathbf{w}$ in the studied MRD and ERV models.
Interactive Displays in Vehicles: Improving Usability with a Pointing Gesture Tracker and Bayesian Intent Predictors

Let \( \bar{e}_k = [x_k, y_k]^T \) be the mouse cursor position at time instant \( t_k \), the predicted cursor endpoint is

\[
\bar{e}_{M} = \bar{e}_1 + d_M [\bar{e}_M - \bar{e}_1] / \| \bar{e}_M - \bar{e}_1 \|_2
\]  \hspace{1em} (2)

where \( d_M \) is the total distance travelled by the cursor from the pointing initial time \( t_1 \) to the finish-time \( t_M \). Such methods leverage cursor movement kinematics where \( d_M \) is a function of the cursor velocity.

\[
d_M = \| \bar{e}_{M} - \bar{e}_1 \|_2 = a \nu_{\text{max}} + b \text{ such that } \nu_{\text{max}} = \max_i \| \bar{e}_i - \bar{e}_{i-1} \|_2 \text{ is a priori learnt peak velocity and } \| x \|_2 = \sqrt{\sum_{i=}^M x^2(n_i)} \text{ is the } L_2 \text{ norm of } x \in \mathbb{R}^M.
\]

the velocity at \( t_k \) is given by:

\[
\nu_k = a \nu_{\text{max}} + b d_M + c.
\]

The set of quadratic coefficients is calculated from the available cursor track and \( d_M \) is extrapolated for \( \nu_M = 0 \). In those methods, the cursor is assumed to head at a nearly constant angle towards the collinear possible targets. As this layout is rarely faced in practice, estimating \( d_M \) does not necessarily establish the destination since a group of selectable items are usually present. The pursued objective here is to reliably establish the GUI icons likelihoods rather than the gesture length as with the regression-extrapolation approach. It is shown below that in a typical 3D pointing task, the angle of travel of the pointing finger can vary drastically overtime and using (2) can lead to misguided predictions.

SET-UP AND POINTING GESTURE CHARACTERISTICS

In this section, we discuss the various modules used to realise the proposed intent inference system whose block diagram and its implementation within an instrumented car are depicted in Fig. 1. The characteristics of the pointing gesture are also examined using in-vehicle data pertaining to a sample of four male passengers of varying ages and heights undertaking a relatively large number of pointing tasks.

Instrumentation and Procedure

Interactive Display and Stimuli

An 11.5 inch Windows 8 tablet mounted to the vehicle dashboard is used to display a GUI for testing purposes. The latter interface is a touchscreen application that displays various layouts of selectable circular items of radii specified by the user. For example, selectable icons can have a circular formation as in Fig. 1a. After filling the user information form, an icon appears on the display, highlighted in red, and the user is required to touch this icon to make a selection. The designed application logs all detected interactions with the touchscreen and their time-stamps, e.g., successful and missed selections.

Pointing Finger/Hand Tracking

With the recent advances in computer vision technology, new commercial 3D vision sensory devices are emerging, e.g., Microsoft Kinect and Leap Motion (LM) controller. They promote gesture-recognition-based HCI Here, we employ a LM controller to continuously track the user’s pointing hand/foot(s) movements as he/she attempts to select an icon on a touchscreen interface. Among other parameters, the LM sensor produces the Cartesian coordinates of the pointing
finger-tip along the \(x\), \(y\) and \(z\) axes at an average rate of 50 observations per second. For example, at time instant \(t_k\), we have \(\mathbf{e}_k \triangleq [x_k, y_k, z_k]^T\) such that \(\mathbf{e}_k\) values (in mm) are relative to the sensor origin and orientation. Each of Figs. 2a and 2b depicts three typical pointing finger-tip full trajectories throughout a pointing task. Data captured in a stationary vehicle and a car moving at varying speeds are shown in Fig. 2a and Fig. 2b, respectively. The interface is located at a fixed distance from the LM along the \(z\) axis (see Fig. 1a). It is noted that the finger traces in the mobile vehicle exhibit high level of fidelity with erratic movements incurred by substantial perturbations introduced by the driving and road conditions.

Inertia Measurement Unit
A high accuracy digital IMU is employed to measure the experienced in-vehicle vibrations and lateral accelerations; data rate is \(\approx 400\) Hz. Define \(\mathbf{a}_k \triangleq [a_x(t_k), a_y(t_k), a_z(t_k)]^T\) and \(\mathbf{g}_k \triangleq [g_x(t_k), g_y(t_k), g_z(t_k)]^T\) as the measured 3D accelerometer and gyroscope data at time \(t_k\), respectively.

Analysis and Prediction
The touchscreen application, LM and IMU data are processed by a software module running on the tablet. Synchronisation between the various data sources is maintained via the tablet. The LM data require sorting and processing, especially that several objects are in the LM field of vision and the sensor often loses track of the pointing finger. The adopted probabilistic intent predictors utilise the logged data to infer early in the pointing task the likelihood of each selectable GUI icon and/or the destination. The prediction outcome can be fed back to the GUI to enable a pointing facilitation method.

Driving and Road Conditions
The instrumented vehicle is driven under various road and driving conditions, which greatly affect the pointing gesture and thereby the success rate of selecting an item on the touchscreen. In Table 1, we show the average success rate of pointing gestures (i.e., whether the correct item is selected) when the vehicle is stationary and mobile. In the latter case, the car is driven over: 1) a motorway at nearly constant speed, 2) road with uneven surface at varying speeds and 3) off-road with significant perturbations. The table clearly demonstrates the impact of the road/driving conditions on the interactive display usability and sheds light on the cognitive/movement/visual capacity required to successfully complete the undertaken selection tasks. The success rates for a mobile vehicle are expected to be tangibly lower than those reported in Table 1 if the test subjects primary task was driving, i.e., they were drivers and not passengers.

Pointing Gesture Characteristics
First, the correlation between the pointing gesture and in-vehicle accelerations is addressed. In principle, we can write

\[
\tilde{\mathbf{a}}_k = \mathcal{F}(\mathbf{a}_k) + \mathbf{e}_k
\]

(3)

where \(\mathbf{e}_k\) is the acceleration from the LM pointing data at \(t_k\), \(\mathcal{F}(\cdot)\) is a generic function, \(\mathbf{a}_k\) is the IMU data incorporating the experienced gravitational as well as dynamic accelerations and \(\mathbf{e}_k\) is some noise factor. Collected data has shown that the correlation between \(\tilde{\mathbf{a}}_k\) and \(\mathbf{a}_k\) can be marginal. This is attributed to the human highly nonlinear behavior in perturbed environments, sitting position, seat cushioning, screen position, etc. Fig. 3 depicts the recorded in-vehicle accelerations for a highly perturbed scenario (driving over road with uneven surface) and its correlation with \(\tilde{\mathbf{a}}_k\), the correlation index is \(c \in [-1, 1]\). The figure shows that the sought correlation is weak and the relationship in (3) can be intractable. Hence, relying on the IMU measurements to compensate for the present noise can be futile. Nonetheless, the IMU data can be used to decide whether an erratic pointing behavior (see Fig. 2b) is expected and accordingly modify the applied pre-processing and/or pointing movements model.

Fig. 4 exhibits the angle between the pointing finger heading direction and the destination, i.e., \(\theta_k \triangleq \angle (\mathbf{e}_k - \mathbf{e}_{k-1}, \mathbf{b}^*)\)

<table>
<thead>
<tr>
<th>Condition</th>
<th>Average No. of tasks</th>
<th>Success rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stationary</td>
<td>40</td>
<td>97 %</td>
</tr>
<tr>
<td>Motorway</td>
<td>170</td>
<td>83 %</td>
</tr>
<tr>
<td>Uneven Road</td>
<td>108</td>
<td>67 %</td>
</tr>
<tr>
<td>Off-road</td>
<td>75</td>
<td>25 %</td>
</tr>
</tbody>
</table>

Table 1: Pointing/selection tasks average success rate.
where $c_{t-1}$ and $c_t$ are two successive finger-tip positions. Vector $b = [b_{x,1}, b_{y,1}, b_{z,1}]^T$ is the 3D coordinates of the intended GUI selectable icon. The displayed data belongs to 30 recorded typical pointing tasks in a stationary vehicle and the instrumented vehicle in Fig. 1 driven over roads of uneven surfaces. The durations of the pointing gestures are in the range of $t_M \in [0.4, 1.8]$ seconds and their lengths are $d_M \in [10, 16]$ cm. Both plots show that $b_k$ values drastically change throughout the selection task, which illustrates the non-optimal nature of the pointing gesture. In the stationary case, $b_k$ tends to be large as the pointing gesture reaches the intended destination. The premise that the destination is along the heading direction of the pointing finger as in (2) does not hold and leads to incorrect predictions. The erratic changes in the $b_k$ values are more prominent in Fig. 4b.

The pointing finger speed, i.e. $|c_k - c_{k-1}|/t_k$, is depicted in Fig. 5 for the same data set in Fig. 4. It can be noticed in Fig. 5a that initially the pointing speed builds-up until it reaches a peak, i.e. quickly converging on the target. It then gradually decays prior to approaching the intended GUI icon. It is noted that unlike pointing in 2D via a mouse, the velocity of the pointing finger is not zero when performing the selection action, i.e. upon touching the screen. Thus, linear-regression techniques that interpolate for a velocity equal to zero to decide on the destination do not apply to the pointing-gesture based interactions in 3D. The speed on contact with the display can notably vary as seen in Fig. 5. In the moving vehicle scenario with perturbations present, the preceding velocity profile is less visible.

**INTENT INFERENCE ALGORITHMS**

At time instant $t_k$, $c_{1:k} = \{c_1, c_2, ..., c_k\}$ comprises all the available pointing finger-tip coordinates at the consecutive discrete-times $\{t_1, t_2, ..., t_k\}$. Let $B = \{b_i: i = 1, 2, ..., N\}$ be the set of $N$ GUI selectable items. No assumptions are made about their distribution and their locations are known where $b_i = [b_{x,i}, b_{y,i}, b_{z,i}]^T$ denotes the coordinates of the centre of the $i^{th}$ icon. Recalling that $t_M \geq t_k$ is the total duration of the pointing task, the objective here is to infer the intended target on the interactive display from $c_{1:k}$. The adopted inference routine consists of the following two steps:
1. Determine the likelihood of all the selectable GUI items, i.e. $P(t_k) = \{P(B_i | \bar{c}_i); i = 1, 2, \ldots, N\}$.

2. Make a decision on the intended destination. This is equivalent to calculating the Maximum a Posteriori (MAP) via

$$\hat{B}(t_k) = \arg \max_{B_i \in \mathcal{B}} P(B_i | \bar{c}_i; t_k)$$

for the set of $N$ nominal targets where $\hat{B}(t_k)$ is the predicted destination and $P(B_i | \bar{c}_i; t_k) \propto P(B_i) P(c_i; B_i)$ according to Bayes' rule. Therefore, the adopted Bayesian approach relies on a belief-based inference followed by a classifier. Since the aim is to utilise the available pointing trajectory to determine the destination, we assume a uniform prior on all items, i.e., $P(B_i) = 1/N$ for $i = 1, 2, \ldots, N$. In this case, the classification problem in (4) corresponds to the maximum likelihood estimation and the solution relies solely on establishing $P(c_i; B_i)$ for $i = 1, 2, \ldots, N$.

It might be desirable to utilise only the last $L$ logged finger positions, i.e., $c_{i-L}, c_{i-L+1}, \ldots, c_i$, and $k - L > 0$, to determine $\hat{B}(t_k)$. A sliding time window is applied to the data and the window width is a design parameter. Next, several simple probabilistic models that enable calculating $P(c_i; B_i)$, $i = 1, 2, \ldots, N$ and hence (4) are introduced and subsequently evaluated.

**Nearest Neighbour (NN) Model**

This is a simple and intuitive approach that chooses the interface selectable icon that is closest to the current position of the pointing finger, i.e., $B_i \in \mathcal{B}$ with the smallest Euclidean distance $d_{k,i} = ||c_k - B_i||_2$, $i = 1, 2, \ldots, N$. In a probabilistic framework, this can be expressed as

$$P(c_k | B_i) = \mathcal{N}(c_k; B_i; \sigma_{B_i}^2).$$

The observation vector $c_k$ has a multivariate normal distribution with a mean equal to that of the possible destination and a fixed covariance $\sigma_{B_i}^2$. The latter is a design parameter. Assuming that the logged finger positions at various time instants are independent for simplicity, the sought $P(c_k; B_i)$ reduces to $P(c_k; B_i) = \prod_{i=1}^{k} P(c_i; B_i)$.

**Bearing Angle (BA) Model**

This is based on the premise that the pointing finger is heading directly towards the intended destination i.e. the cumulative angle between the finger positions and the target is minimal. For every two consecutive measurements, the bearing angle with respect to the destination can be assumed to be a random variable with zero mean and fixed variance as per

$$P(c_k | c_{k-1}, B_i) = \mathcal{N}(\theta_{k,i}; 0, \sigma_{\theta_{B_i}}^2)$$

where $\theta_{k,i} = \angle(v_k, b_i)$ for $B_i, v_k = c_k - c_{k-1}$ and $\sigma_{\theta_{B_i}}^2$ is a design parameter. We can then write

$$P(c_k; B_i) = P(c_1; B_i) \prod_{i=2}^{k} P(c_i | c_{i-1}, B_i).$$

This algorithm can be considered to represent the best outcome of the linear-regression-extrapolation techniques; e.g.
assuming that $d_M$ in (2) is accurately estimated. According

to (6) and (7), BA forms a wedge-shaped confidence interval

whose width is set by $\sigma_B^2$. Any selectable icon that falls

within this region is assigned a high probability. With many

selectable targets in close proximity from one another, the

possibility that the BA model leading to an erroneous pre-
diction is high. Additionally, as the pointing finger ap-

proaches the intended destination, $d_k$ can become large, e.g.

see Fig. 4, undermining the inference confidence level.

**Heading and Solid Angle (HSA) Model**

An object $B_i$ has a smaller solid angle if the observer is far from its location

compared with that if the observer is nearby as demonstrated in Fig. 6. Solid angle (in steradians) of a sphere located at
distance $d_{i,k}$ is approximated by

$$\Omega_{i,k} = \frac{A \cos(\alpha_k)}{a^2 - d_{i,k}^2 + A}$$

where $A$ is the area of the target object. Targets of arbi-

trary shapes can be closely approximated by a number of

spheres. Parameter $\alpha_k$, which is the exposure angle, is ir-

relevant to the prediction problem and $\alpha_k = 0$ is assumed.

The direction of travel is specified by the measured velocity

vector $v_k$ at $t_k$ and the HSA likelihood probability for two

consecutive pointing positions can be approximated via

$$P(\mathbf{c}_k|\mathbf{c}_{k-1}, B_i) = \mathcal{N}(\mathbf{c}_k, \mathbf{\Sigma}_{i,k}, \mathbf{\Gamma}_k^2)$$

By applying (12) and (14) into (7), the likelihood of the possible

GUI icons are attained prior to unveiling the intended destina-
tion via the MAP estimator in (4); equal priors are assumed.

Given the linear nature of (11) and assuming a linear meas-

urements equation, e.g. $\mathbf{m}_k = \mathbf{H}_k \mathbf{c}_k + \mathbf{n}_k$, the MRD model can be implemented using a bank of $N$ Kalman filters that combine the prediction with the trajectory-smoothing opera-

tion. This is outside the scope of this paper.

**EVALUATION AND RESULTS**

Here, we evaluate the performance of the proposed prediction approach on 52 data sets of typical pointing tasks conducted in a stationary and mobile vehicle. In each experiment, the user selects a target on the interface by reaching to the tablet touchscreen as in Figs. 1 and 2. The layout of the GUI is similar to that shown in Fig. 1a and $N = 44$ selectable targets are present (less than 1.5 cm apart) in lieu of 19. The intent prediction is performed at the arrival of each new LM

observation indicating the pointing finger coordinates. The
design parameters of the inference models, e.g. $L$, are chosen such that the highest likelihood probabilities $P(\mathbf{c}_k|B^*)$ are

prioritized ($B^*$ is the true destination on the display).

We assess the performance of a predictor in terms of its clas-
sification success from the MAP estimator in (4), i.e. when

or how early in the pointing gesture the predictor assigns the

highest probability to the true intended target $B^*$. This is de-

scribed in Fig. 7 against the percentage of completed pointing
gesture (in time) and averaged over all the conducted pointing
tasks. It can be noticed from the figure that mean rever-

tering diffusion model provides the earliest successful pre-
dictions of the intended destination, especially in the crucial

first 85% of the pointing gesture. The performance of NN

and MRD are similar in the final portion of the pointing task

since the user’s finger is inherently close to the intended item

at this stage, i.e. briefly before the selection action. In prac-
tice, an early prediction, e.g. in the first 58% of the pointing
task duration, is more effective at minimising the user visu/
movement/cognitive effort, enabling early pointing facilitation techniques and enhancing the overall user experience. The benefits of successful intent inference in the last 15% of the pointing gesture duration are limited since the user has already dedicated the necessary attention to execute the selection task. The BA model outcome exhibits pronounced fluctuations due to the continuously changing heading angle as shown in Fig. 4, especially towards the end of the selection task. Overall, Fig. 7 demonstrates that the proposed approach can remarkably minimise the pointing gesture duration/effort where successful predictions can be made as early as 10% into the pointing gesture.

In Fig. 8, we depict the proportion of the total pointing gesture (in time) the algorithms correctly predicted the true intended item on the interface. It is clear from the figure that the MRD model consistently delivers higher overall success rates compared with NN, BA and HSA models. For example, MRD infers the correct intended destination in approximately 65% of the performed predictions or the pointing gesture duration compared to 54% with the nearest neighbour approach. HSA brings tangible benefits compared to BA and its performance is comparable to the NN.

These results illustrate the notable gains furnished by the introduced pointing-gesture-based approach in terms of reducing the pointing duration/effort. It is important to note that interactions with displays are very prevalent in modern vehicle environments. Therefore, small improvements in the pointing task efficiency, even reducing the pointing time by few milliseconds, will have substantial aggregate benefits on safety and the overall user experience, especially for a driving user.

CONCLUSIONS AND DISCUSSION
In this paper, we proposed a novel framework to improve the interactive displays usability in the vehicle environment. We deployed a computer-vision tracking device in conjunction with Bayesian intent predictors. It is shown that the simple adopted computationally efficient prediction models can provide significant performance enhancements by substantially reducing the duration of the pointing task. This minimises the required time, physical effort, visual attention and cognition to interact with an in-vehicle infotainment system. Any available priors gathered from relevant contextual information can be easily leveraged, e.g. target selection history, interface design or layout, user profile or behavior, etc. In principle, the predictors spatial region(s) of interest can extend to far beyond the touchscreen platform, e.g. areas in the vicinity of user seat, and can support in-vehicle virtual displays.

Perturbed pointing trajectories can be smoothed using classical linear filtering methods in a pre-processing stage, i.e. prior to the prediction operation. However, the highly non-linear nature of the observed tracks as in Fig. 2b can render such an approach ineffective. Alternatively, advanced Bayesian filtering algorithms, e.g. sequential Monte Carlo techniques [8], can be applied. They can better model such processes that incorporate erratic movements/jumps due to adverse road and driving conditions. Nevertheless, computationally intensive approaches that levy prohibitively large delays can hinder real-time implementations and lead to impractical pointing assistive regimes. This trade-off should be closely assessed when adopting advanced filtering algorithms.

This paper serves as an impetus to further research and calls for a complete experimental evaluative study. The latter should consider a large number of subjects, human performance factor and categorise the road as well as the driving conditions to best quantify the gains of the intent predictors and/or choose the form(s) of subsequent pointing facilitation technique(s) based on the user experience.

Figure 8: Percentage of the pointing gesture (in time) the destination inference method made a correct prediction.
Filtering Perturbed In-Vehicle Pointing Gesture Trajectories: Improving the Reliability of Intent Interference

In this paper, we propose a low-complexity online Sequential Monte Carlo (SMC) filtering technique that can effectively track and thereby eliminate/suppress the erratic unintentional movements, e.g., due to road/driving conditions. It is based on appropriately modelling the perturbed pointing trajectory. The introduced filtering is a pre-processing operation that precedes and is independent of the intent inference algorithm as shown in Fig. 3. It is noted that the filtering task is performed online, i.e., as each LM data frame arrives indicating the observed pointing finger location. Accordingly, an efficient implementation of the proposed particle filter is also addressed below.

2. PROBLEM FORMULATION

At time instant $t_k$ the LM controller produces Cartesian coordinates of the pointing finger $m_k = [x_k, y_k, z_k]^T$ and $m_k$ values (m/mm) are relative to the sensor origin as well as orientation. Thus, $m_{1:k} = \{m_1, m_2, \ldots, m_k\}$ comprises all the available pointing finger-up coordinates at consecutive discrete observation times $\{t_1, t_2, \ldots, t_k\}$. Let $B = \{B_i : i = 1, 2, \ldots, N\}$ be the set of N GUI selectable items. Their locations are known, given by $b_i = [b_{ix}, b_{iy}, b_{iz}]^T$. The intent inference aims to predict the intended destination, i.e., $B^*$, on the GUI, using the first $k$ observations $m_{1:k}$.

Due to road and driving conditions, we assume that the recorded pointing finger movements are given by a combination of intention- and perturbation-related movements, e.g.,

$$m_k = c_k + p_k + e_k$$  \hspace{1cm} (1)

where $c_k = [x_k, y_k, z_k]^T$ is the location of the pointing finger after removing/suppressing the unintentional perturbations-related movements $p_k$, and the measurement noise is denoted by $e_k$. The objective of this work is to extract $c_k$ from the LM observations $m_{1:k}$, with the aim of improving the prediction quality of the intent inference module. After analysing the salient features of the pointing gesture, namely the velocity profile, we propose a nonlinear model and a low-complexity SMC filtering technique to achieve the perturbations elimination/suppression.

3. TRAJECTORY MODELLING

In a vehicle environment the motion of a user’s finger will differ from their true intention due to jolts and jerks arising from road bumps and vehicle motion, especially on uneven road surfaces. Fig. 4 shows the velocity profile of a number of pointing tracks in a moving vehicle, illustrating a noisy process subject to large non-stationary, but zero-reverting deviations. A model is proposed that divides the observed motion into two constituent components: that due to the intended pointing gesture, and that due to the perturbations caused by external motion. The path of the intended pointing gesture $c$ is assumed to be smooth and is therefore modelled as a near constant velocity motion model with low velocity noise:

$$dc = \sigma_v dW_t$$

$$dc = \epsilon dt,$$

where $dW_t$ is the instantaneous change in a standard Brownian motion. It would be straightforward to use other linear motion models for the intended gesture, for example mean-reverting
models such as those used for intentionality prediction in section 5.

Perturbations are modelled as having a jump-diffusion driven, mean-reverting velocity process, governed by
\[ dp = \sigma_p dW_2 + \sigma_p dJ - \lambda_1 \dot{p} dt \]
where \( dJ \) is the instantaneous change in the jump process.
\[ J_t = \sum_{i=1}^{\tau_t} P_i \]
with \( P_i \sim \mathcal{N}(0,1) \) and where \( \tau_t \) is the number of jumps observed in the interval \([0, t]\). Jump times are assumed to be exponentially distributed with rate \( \tau_t \), with the distribution of the next jump time \( \tau \) given by the memoryless distribution
\[ p_{\text{exp}}(\tau) = \text{Exponential}(\tau_t). \]
The effect of this is to allow occasional large impulsive shocks to the perturbation velocity, allowing sharp jolts to be modelled.

The overall finger motion process \( f \) is then given by integration of the velocity components arising from the intention \( \dot{x} \) and from the perturbation \( \dot{p} \). The motion is also assumed to revert towards the intended motion, so that shocks caused by perturbations are not persistent. Therefore,
\[ df = (\dot{p} + \dot{c}) dt - \lambda_2 (f - \dot{c}) dt. \]
It is this \( f \) process that is assumed to be (noisily) observed by the LM sensor, so that the \( k \)th measurement at observation time \( t_k \) is given by
\[ m_k = f_k + \varepsilon_k, \]
with \( \varepsilon_k \sim \mathcal{N}(0, \sigma^2_{\text{obs}}). \) For the LM controller, sensor noise levels, specified by \( \sigma^2_{\text{obs}} \) are small compared to the scale of the \( f \) process.

In matrix-vector form the system for a single spatial dimension \( d \in \{x, y, z\} \) can be written as
\[ dX = AX dt + BdW + CdJ \]
with \( dW = [dW_1, dW_2]^T \) and
\[ X = \begin{bmatrix} \varepsilon_d \\ \varepsilon_d \\ \varepsilon_p \\ \lambda_2 \\ 1 - \lambda_2 \\ 1 \end{bmatrix}, \quad A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\lambda_1 \\ \lambda_2 & 1 - \lambda_2 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ \sigma_c \\ 0 \\ 0 \end{bmatrix}, \quad C = \begin{bmatrix} 0 & 0 & \sigma_c & 0 \end{bmatrix}^T \]
This is extended to three spatial dimensions by assuming the dimensions are independent, giving a 12-dimensional system, with block-diagonal \( A, B \) and \( C \) matrices.

Excluding jumps, the system is a standard linear time-invariant (LTI) system, with the distribution of \( X_t \) given by a Gaussian with, for \( s < t \),

\[ \mathbb{E}(X_t) = e^{A(t-s)} \mathbb{E}(X_s) \]
\[ \text{cov}(X_t) = e^{A(t-s)} \left[ Q(s,t) + \text{cov}(X_s) \right] e^{A(t-s)^T} \]
where
\[ Q(t,s) = \int_s^t e^{A(s-r)} BB^T e^{A(t-r)^T} dr \]
This latter integral can be found using the Matrix Fraction Decomposition.

The presence of zero-mean Gaussian jumps does not affect the calculation of the process mean, so (2) can be used to find it at any time, regardless of jump positions. The effect of jumps on process covariance is most easily calculated by considering the process piecewise, consisting of diffusion-only periods between jumps, followed by a jump which serves to instantaneously inflate the covariance (additively). This increased covariance is then propagated through subsequent diffusion periods using (3).

4. FILTERING IMPLEMENTATION

The presence of jumps in the model in section 3 means that the Kalman filter cannot be used directly for state inference. However, conditional on the jump times, the system is linear Gaussian, meaning that an efficient Rao-Blackwellized particle filter can be used for inference, in which the inferred state is divided into non-Gaussian jump-times \( T_k \) and the remaining state, including user intention, which is linear-Gaussian. Each particle therefore consists of a set of jump time \( T_k \) giving jump times observed up to the observation.
time $t_k$. Conditioned on this set of jump times sampled in each particle, the remaining state can be inferred using a Kalman filter. Alg. 1 shows a single step of the particle filter, which allows a weighted particle collection representing the distribution of jump times given the first $k - 1$ observations to be updated to incorporate information from the $k$th observation.

**Algorithm 1** Single step of VRPF algorithm (outline)

Require: Collection of $N_{k-1}$ weighted particles $(w_{k-1}^{(i)}, T_{k-1}^{(i)})_{i=1}^{N_{k-1}}$ representing $p(T_{k-1} \mid m_{i(k-1)});$

target number of particles $N_k$

for (each particle) $i = 1 \ldots N_{k-1}$ do
  Choose number of offspring: $N_{k}^{(i)} \sim R(\cdot \mid w_{k-1}^{(i)} N_{k-1}^{-1})$
end for

Set particle index $p := 1$

for (each particle) $i = 1 \ldots N_{k-1}$ do
  $N_{k}^{(i)} = 0$ (number of non-jumping offspring)
  for (each offspring) $j = 1 \ldots N_{k}^{(i)}$ do
    Sample jump time $\tau \sim p_{jump}(\cdot)$
    if $\tau < t_k$ then
      $T_{k}^{(i,j)} := T_{k-1}^{(i,j)} + \tau; v_k^{(i,j)} := \frac{1}{N_{k}} \sum_{j=1}^{N_{k}^{(i)}} \sigma_{k}^{(i,j)}; p := p + 1$
    else
      $N_{k}^{(i)} := N_{k}^{(i)} + 1$
    end if
  end for
  if $N_{k}^{(i)} > 0$ then
    $T_{k}^{(i)} := T_{k-1}^{(i)} + \frac{1}{N_{k}} \sum_{j=1}^{N_{k}^{(i)}} \sigma_{k}^{(i,j)}; p := p + 1$
  end if

end for

$N_k := p - 1$

for (each new particle) $i = 1 \ldots N_k$ do
  Calculate $\mu_k^{(i)}, \Sigma_k^{(i)} = \text{Kalman-Filter}(m_k, T_k^{(i)})$
  $w_k^{(i)} := w_{k-1}^{(i)} p(m_k \mid m_{i(k-1)}, T_k^{(i)})$
end for

Normalize weights: $w_k^{(i)} := w_k^{(i)} / \sum_i w_k^{(i)}$

return Collection of $N_k$ weighted particles $(w_k^{(i)}, T_k^{(i)})_{i=1}^{N_k}$ representing $p(T_k \mid m_{i(k)})$

Since the non-Gaussian portion of the state consists only of jump times, a further computational efficiency gain can be made by using the Variable Rate Particle Filter (VRPF) algorithm of [15, 16]. This exploits the observation that all offspring of a given particle having no proposed jump between $t_{k-1}$ and $t_k$ are identical, and can therefore be represented by a single over-weighted particle, with weight proportional to the number of non-jumping offspring.

In Alg. 1, the resampling function $R(\cdot \mid w_{k-1}^{(i)} N_{k-1}^{-1})$ is a function that returns a number of offspring proportional to the particle weight. Here, we use a residual sampling scheme, but other sampling schemes would also be possible.

Because jumps are rare events (i.e. the jump rate $r_j$ is small compared to the observation timestep), sampling jumps from their prior $p_{jump}(\tau)$ will be inefficient, as many samples will be required before a jump in the interval $[t_{k-1}, t_k]$ is drawn. Therefore, in this work, a uniform jump proposal distribution is used, which proposes jumps in $[t_{k-1}, t_k \pm q_k (t_k - t_{k-1})]$, with $q_k > 1$ (here $q_k = 4$).

The Kalman filter step, (Kalman-Filter $(m_{i(k)}, T_k^{(i)})$ in Alg. 1), uses a Kalman filter to obtain the mean $\mu_k$ and covariance $\Sigma_k$ for the linear Gaussian portions of the state corresponding to the jump sequence represented by particle $i$. Over the entire particle collection, the distribution of these linear-Gaussian states (including user intentionality), is given as a Gaussian mixture. The observation likelihood $p(m_k \mid m_{i(k-1)}, T_k^{(i)})$ is given by the Prediction Error Decomposition from the Kalman filter. Both this and the state mean and covariance can be calculated in a single Kalman filter step if the previous mean and covariance for each particle is stored at each stage.

5. DESTINATION INFERENCE

The objective of destination inference is to predict the intended target on an interactive display using $e_{i,k}$, when $t_k \leq t_M$, the total duration of the pointing gesture. Each inference routine determines the likelihood of all $N$ selectable GUI items, i.e. $P(B_i) = \{p(c_{i,k} \mid B_i), i = 1, 2, \ldots, N\}$ and then makes a decision on the intended destination. This is equivalent to calculating the maximum a posteriori (MAP) estimate assuming a uniform prior, i.e. $p(B_i) = 1/N$ for $i = 1, 2, \ldots, N$. In some cases, it is desirable to use only the last $L$ logged finger positions to determine $\hat{B}(t_k)$ with $L$ a design parameter.

- **Nearest Neighbour (NN):** this is an intuitive models that assigns the highest probability to the target closest to the current finger position. The likelihood of $B_i$ can be expressed as $p(c_{i,k} \mid B_i) = N(0, \sigma_{B_i}^2)$ where $\sigma_{B_i}^2$ is covariance of the multivariate normal distribution. Assuming independent observations for simplicity, we have $p(c_{i,k} \mid B_i) = \prod_{k=L}^{k} p(c_{i,k} \mid B_i)$.

- **Bearing Angle (BA):** this is based on the premise that the user points directly towards the destination. The BA from two consecutive positions can be assumed to be a zero mean random variable with fixed variance, thus $p(c_{i,k} \mid B_i) = N(0, \sigma_{B_i}^2)$ where $\sigma_{B_i}^2$ is a design parameter. The likelihoods are given by: $p(c_{i,k} \mid B_i) = \prod_{k=L}^{k} p(c_{i,k} \mid B_i)$.

- **Mean Reverting Diffusion (MRD):** the pointing
movements are modelled as a multivariate Ornstein-Uhlenbeck process with a mean-reverting term. For the \( y^{th} \) selectable interface icon, we have

\[
c_{i,k} = e^{-\Lambda t} c_{i,k-1} + \left[ I_k - e^{-\Lambda t} \right] b_i + v_{i,k}
\]

where the square matrix \( \Lambda \) sets the mean reversion rate that steers the evolution of the process and \( v_{i,k} \sim \mathcal{N}(0, \sigma^2_{M RD}) \) is an additive Gaussian noise; \( c_{i,k} \) and \( c_{i,k-1} \) are the state vectors with respect to \( B_i \) at the time instants \( t_k \) and \( t_{k-1} \), respectively; and \( \tau_k = t_k - t_{k-1} \). It follows that the distribution of the conditional state is \( p(c_{i,k} | c_{i,k-1}, B_i) = \mathcal{N}(c_{i,k} | \Sigma_{i,k}, \Gamma^2_i) \) such that \( \Sigma_{i,k} = e^{-2\Lambda \tau_k} \Sigma_{i,k-1} + \left[ I_k - e^{-2\Lambda \tau_k} \right] b_i b_i^T \sigma^2_{M RD} \) and \( \Gamma^2_i = \frac{1 - e^{-2\Lambda \tau_k}}{2\Lambda} \sigma^2_{M RD} \); \( \sigma^2_{M RD} \) is process dispersion matrix.

6. RESULTS

Here, we evaluate the performance of the NN, BA and MRD prediction models with and without the proposed SMC filtering approach applied as a pre-processing step. 10 data sets of typical pointing tasks are used, collected in a vehicle being driven over hush terrain. In each experiment, the user selects a target on the interface by reaching to a touchscreen mounted on the vehicle dashboard. The layout of the GUI is similar to that shown in Figs. 2 and 5 with 37 selectable interface items. The intent prediction is performed at the arrival of each new LM observation indicating the pointing finger coordinates. The design parameters of the inference models, e.g., \( L \), \( \sigma_{BN}, \sigma_{BA}, \Lambda \) and \( \sigma^2_{M RD} \) are chosen such that the highest likelihoods \( P(c_{i,k} | B_i) \) are obtained, where \( B_i \) is the true destination on the display. The particle filter was run with 100 particles for the experiments shown, but proved stable in almost all cases with fewer than 10 particles owing to the efficient Rao-Blackwellized VRPF used.

Fig. 5 shows a highly perturbed pointing trajectory before and after filtering. It can be seen here that the jump-diffusion based filter model successfully suppresses the unintentional jolts and maneuvers introduced due to the road/driving conditions. The proportion of the total pointing gesture (in time) for which each of the predictors correctly predicted the true intended item on the interface, with and without filtering. Fig. 6a shows the average combined results for all data sets, whilst Fig. 6b shows the performance improvement (in \% points) due to filtering for each data set and intentionality prediction method. The proposed SMC filtering pre-processing step improves the average prediction accuracy for all inference methods, with minor improvements for the NN and MRD methods, but a substantial improvement for the BA method, which is seen in all experiments. This result is significant because the NN and, to some extent, MRD methods rely on physical proximity to the correct button to make accurate

(a) Mean pointing accuracy

(b) Improvement in accuracy (% points) using filter for each for each pointing task with each prediction method

Fig. 6. Percentage of the gesture (in time) the destination inference method made a correct prediction.
predictions (e.g. for the NN method the Voronoi region of each button in the interface can be extruded to a prism perpendicular to the screen within which that button is nearest), and thus can perform poorly for tracks moving at an angle to the interface surface. On the other hand, the BA method takes account of the direction of travel of the pointing finger and so is better able to cope with such situations. Previously, however, the BA method was found to perform badly compared to the other methods considered for perturbed tracks. This was due to the instability of the track heading in the presence of perturbations. By applying filtering as shown here the performance of this algorithm can be made comparable with the previous best algorithm, MRD, making it a viable alternative (or component within) intentionality inference, likely to work in situations where other algorithms will fail.

In the tracks considered here the presence of jumps was inferred in about 2-4% of observation periods, suggesting a handful of jumps in most tracks. In tracks from less harsh road conditions, however, it is possible that no jumps will be present in some cases. In such cases a pure-diffusion system is likely to produce comparable results, allowing efficient computation with a Kalman filter.

7. CONCLUSIONS

This paper outlines a Bayesian filtering framework for extracting intended user pointing gestures in automotive environments, even in the presence of large perturbations arising from uneven road surfaces. The proposed filtering method showed particular benefits when used with bearings-based intentionality prediction methods, allowing such methods to perform comparably with state-of-the-art methods for the tracks tested. The filtering method presented thus opens up the possibility of including such bearings-based methods as a useful component in an intentionality prediction system.

Interactions with displays are very prevalent in modern vehicle environments, so small improvements in pointing task efficiency, even reducing the pointing time by few milliseconds by improving the prediction quality, will have substantial aggregate safety and user experience benefits, especially for a driving user. A more complete experimental study that considers a large number of subjects is needed to best quantify the gains of SMC-based filtering, but the results shown here illustrate that such filtering can offer substantial benefits when used as a pre-processing step for certain types of intentionality inference algorithms, without degrading the performance of others.
CLAIMS

1. A method of determining an intended target of an object, comprising:

   determining a location of the object at a plurality of time intervals;

   determining a metric associated with each of a plurality of targets, the
   metric indicative of the respective target being the intended target of the
   object, wherein the metric is determined based upon a model and the
   location of the object at the plurality of time intervals;

   determining, using a Bayesian reasoning process, the intended target from
   the plurality of targets based on the metric associated with each of the
   plurality of targets.

2. The method of claim 1, comprising determining a trajectory of the object.

3. The method of claim 2, wherein the trajectory of the object comprises data
   indicative of the location of the object at a plurality of time intervals.

4. The method of claim 2 or 3, comprising filtering the trajectory of the object.

5. The method of claim 4, wherein the filtering smooths the trajectory of the
   object and/or the filtering reduces unintended movements of the object and/or
   noise from the trajectory.

6. The method of any preceding claim, wherein the model is a Bayesian
   intentionality prediction model.

7. The method of any preceding claim, wherein the model is a linear model.

8. The method of claim 7, wherein the linear model is based on one or more
   filters; optionally the one or more filter are Kalman filters.

9. The method of any of claims 1 to 6, wherein the model is a non-linear model.

10. The method of claim 9, wherein the non-linear model incorporates irregular
    movements of the object.
11. The method of claim 9 or 10, wherein the non-linear model is based on one or more particle filters.

12. The method of any preceding claim, wherein the model is a nearest neighbour (NN) model.

13. The method of claim 12 wherein the NN model determines the metric based upon a distance between the location of the object and each of the targets.

14. The method of claim 12 or 13 wherein the metric is indicative of a distance between the object and each of the targets.

15. The method of any of claims 1 to 11, wherein the model is a bearing angle (BA) model.

16. The method of claim 15 when dependent upon claim 2 or any claim dependent thereon, wherein the metric is indicative of an angle between the trajectory of the object and each of the targets.

17. The method of any of claims 1 to 11, wherein the model is a heading solid angle (HSA) model.

18. The method of claim 17, wherein the metric is indicative of a solid angle between the object and each of the targets.

19. The method of any of claims 1 to 11, wherein the model is a Linear Destination Reversion (LDR) or a Nonlinear Destination Reversion (NLDR) model.

20. The method of claim 19, comprising determining a model for each of the targets.

21. The method of claim 20 when dependent upon claim 2 or any claim dependent thereon, wherein the metric is indicative of the model best matching the trajectory of the object.

22. The method of claim 19 when dependent upon claim 2 or any claim dependent thereon, wherein the NLDR model comprises non-linear perturbations of the trajectory.
23. The method of any of claims 1 to 11, wherein the model is a Mean Reverting Diffusion (MRD) model.

24. The method of claim 23, wherein the MRD models a location of the object as a process reverting to the intended target.

25. The method of any preceding claim, wherein the model is an Equilibrium Reverting Velocity (ERV) model.

26. The method of claim 25, wherein the metric is based upon a speed of travel of the object to the target.

27. The method of any of claims 19 to 26, comprising determining a state of the object.

28. The method of any preceding claim comprising receiving one or more items of environmental information.

29. The method of claim 28, wherein the environmental information comprises one or more of: information indicative of acceleration, information indicative of a state of the vehicle and/or image data indicative of vehicle surroundings.

30. The method of claim 28 or 29, wherein the determination of the metric is based, at least in part, on the one or more items of environmental information.

31. The method of claim 28, 29 or 30, wherein the model is selected based, at least in part, on the one or more items of environmental information.

32. The method of any preceding claim, wherein the determining the intended target is based on a cost function.

33. The method of claim 32, wherein the cost function imposes a cost for incorrectly determining the intended target.

34. The method of claim 32 or 33, wherein the intended target is determined so as to reduce the cost function.
35. The method of any preceding claim, wherein the determining the intended target is based on one or more items of prior information.

36. The method of claim 35 wherein the prior information is associated with at least some of the targets.

37. The method of any preceding claim, comprising selecting a plurality of most recent time intervals, wherein the determining the metric associated with each of the plurality of targets is based upon the location of the object at the plurality of most recent time intervals.

38. The method of any preceding claim, wherein the object is a pointing object.

39. The method of any preceding claim, wherein the location of the object is determined in three dimensions.

40. The method of any preceding claim, wherein determining the location of the object comprises tracking the location of the object.

41. The method of any preceding claim, wherein determining the location of the object comprises receiving radiation from the object.

42. The method of any preceding claim, comprising outputting an indication of the intended target.

43. The method of claim 42, wherein the indication of the intended target comprises identifying the intended target; optionally the intended target is visually identified.

44. The method of claim 42 or 43, comprising outputting the indication of the intended target and one or more possible targets.

45. The method of any preceding claim, comprising activating the intended target.

46. The method of any preceding claim, wherein the plurality of targets comprise one or more of graphically displayed items or physical controls.

47. The method of any preceding claim, wherein the location of the object is determined in three-dimensions.
48. A system for determining an intended target of an object, comprising:

location determining means for determining a location of the object;

a memory means for storing data indicative of the location of the object at one or more instants in time;

a processing means arranged to:

10 determine a metric associated with each of a plurality of targets of the respective target being the intended target of the object, wherein the metric is determined based upon a model and the location of the object at the plurality of time intervals;

15 determine, using a Bayesian reasoning process, the intended target from the plurality of targets based on the metric associated with each of the plurality of targets.

49. The system of claim 48, wherein the processing means is arranged to perform a method as claimed in any of claims 2 to 47.

50. The system of claim 48 or 49, wherein the location determining means comprises means for receiving radiation from the object.

51. The system of claim 48, 49 or 50, wherein the location determining means comprises one or more imaging devices.

52. The system of any of claims 48 to 51, wherein location data indicative of the location of the object at each instant in time is stored in the memory means.

53. The system of any of claims 48 to 52, comprising one or more accelerometers for outputting acceleration data.

54. The system of any of claims 48 to 53, comprising a display means for displaying a graphical user interface (GUI) thereon, wherein the plurality of targets are GUI items.
55. The system of any of claims 48 to 54, wherein the processing means is arranged to receive environmental data from one or more sensing means; optionally the sensing means comprise means for determining a state of the vehicle and/or imaging devices.

56. A vehicle comprising a processing device arranged, in use, to perform a method according to any of claims 1 to 47 or comprising the system of any of claims 48 to 55.

57. A method, system or vehicle substantially as described hereinbefore with reference to the accompanying drawings.
### Patents Act 1977: Search Report under Section 17

#### Documents considered to be relevant:

<table>
<thead>
<tr>
<th>Category</th>
<th>Relevant to claims</th>
<th>Identity of document and passage or figure of particular relevance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y</td>
<td>28-34 &amp; 55</td>
<td>US 2013/194193 A1 (HONEYWELL INT INC [US]) see e.g. the abstract, paragraphs 4 &amp; 32</td>
</tr>
<tr>
<td>A</td>
<td>-</td>
<td>US 2014/078044 A1 (ALPS ELECTRIC CO LTD) see e.g. the abstract</td>
</tr>
<tr>
<td>A</td>
<td>-</td>
<td>US 2011/083089 A1 (MICROSOFT CORP) see e.g. the abstract, paragraphs 5, 26-28 &amp; 33</td>
</tr>
</tbody>
</table>

#### Categories:

<table>
<thead>
<tr>
<th>X</th>
<th>Document indicating lack of novelty or inventive step</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y</td>
<td>Document indicating lack of inventive step if combined with one or more other documents of same category. &amp; Member of the same patent family</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>A</th>
<th>Document indicating technological background and/or state of the art.</th>
</tr>
</thead>
<tbody>
<tr>
<td>P</td>
<td>Document published on or after the declared priority date but before the filing date of this invention.</td>
</tr>
<tr>
<td>E</td>
<td>Patent document published on or after, but with priority date earlier than, the filing date of this application.</td>
</tr>
</tbody>
</table>

#### Field of Search:

Search of GB, EP, WO & US patent documents classified in the following areas of the UKC:

Worldwide search of patent documents classified in the following areas of the IPC

G06F

The following online and other databases have been used in the preparation of this search report

WPI, EPDOC, INSPEC, XPIPECOM

#### International Classification:

<table>
<thead>
<tr>
<th>Subclass</th>
<th>Subgroup</th>
<th>Valid From</th>
</tr>
</thead>
<tbody>
<tr>
<td>G06F</td>
<td>0003/041</td>
<td>01/01/2006</td>
</tr>
<tr>
<td>G06F</td>
<td>0003/01</td>
<td>01/01/2006</td>
</tr>
</tbody>
</table>