


RESEARCH

Open Access



Dead-reckoning animal movements in R: a reappraisal using *Gundog.Tracks*

Richard M. Gunner^{1*} , Mark D. Holton¹, Mike D. Scantlebury², O. Louis van Schalkwyk^{3,4,5}, Holly M. English⁶, Hannah J. Williams⁴, Phil Hopkins¹, Flavio Quintana⁷, Agustina Gómez-Laich⁸, Luca Börger^{1,9}, James Redcliffe¹, Ken Yoda¹⁰, Takashi Yamamoto¹¹, Sam Ferreira¹², Danny Govender¹², Pauli Viljoen¹², Angela Bruns¹³, Stephen H. Bell², Nikki J. Marks², Nigel C. Bennett¹⁴, Mariano H. Tonini¹⁵, Carlos M. Duarte¹⁶, Martin C. van Rooyen¹⁴, Mads F. Bertelsen¹⁷, Craig J. Tambling¹⁸ and Rory P. Wilson¹

Abstract

Background: Fine-scale data on animal position are increasingly enabling us to understand the details of animal movement ecology and dead-reckoning, a technique integrating motion sensor-derived information on heading and speed, can be used to reconstruct fine-scale movement paths at sub-second resolution, irrespective of the environment. On its own however, the dead-reckoning process is prone to cumulative errors, so that position estimates quickly become uncoupled from true location. Periodic ground-truthing with aligned location data (e.g., from global positioning technology) can correct for this drift between Verified Positions (VPs). We present step-by-step instructions for implementing Verified Position Correction (VPC) dead-reckoning in R using the tilt-compensated compass method, accompanied by the mathematical protocols underlying the code and improvements and extensions of this technique to reduce the trade-off between VPC rate and dead-reckoning accuracy. These protocols are all built into a user-friendly, fully annotated VPC dead-reckoning R function; *Gundog.Tracks*, with multi-functionality to reconstruct animal movement paths across terrestrial, aquatic, and aerial systems, provided within the Additional file 4 as well as online (GitHub).

Results: The *Gundog.Tracks* function is demonstrated on three contrasting model species (the African lion *Panthera leo*, the Magellanic penguin *Spheniscus magellanicus*, and the Imperial cormorant *Leucocarbo atriceps*) moving on land, in water and in air. We show the effect of uncorrected errors in speed estimations, heading inaccuracies and infrequent VPC rate and demonstrate how these issues can be addressed.

Conclusions: The function provided will allow anyone familiar with R to dead-reckon animal tracks readily and accurately, as the key complex issues are dealt with by *Gundog.Tracks*. This will help the community to consider and implement a valuable, but often overlooked method of reconstructing high-resolution animal movement paths across diverse species and systems without requiring a bespoke application.

Keywords: Animal behaviour, Animal movement, Global Positioning System, R (programming language), Track integration, Tri-axial accelerometers, Tri-axial magnetometers

Background

Reconstructing animal movement paths is an important tool in ecology, providing insights into animal space-use, behaviour and habitat selection [1–3]. However, accurate estimation of paths at fine temporal scales has proved a

*Correspondence: richard.m.g@hotmail.com

¹ Swansea Lab for Animal Movement, Department of Biosciences, College of Science, Swansea University, Singleton Park, Swansea SA2 8PP, Wales, UK

Full list of author information is available at the end of the article



persistent challenge [4, 5]. Dead-reckoning is a method used to reconstruct animal movement paths, based on sequentially integrating the vector of travel from a pre-determined position using estimates of heading (also termed ‘bearing’ or ‘yaw’) and speed (and displacement about the vertical axis for 3-D movements), over an elapsed time interval [6–9]. In its most advanced form, it can provide positional data with sub-second resolution, irrespective of the environment [e.g., 10, 11, 12] and it therefore has huge potential for providing data that can elucidate many fundamental behavioural and ecological issues related to space-use.

The concept of dead-reckoning (also termed ‘track integration’) originated to aid nautical navigation [6, 13], though its utility to reconstruct uninterrupted fine-scale (in time and space) animal movement paths by integrating different sensors in animal-attached tags was suggested over three decades ago [14, 15]. Today, this typically involves the simultaneous deployment of tri-axial accelerometers and magnetometers [e.g., 9, 10, 16, 17–20], utilising the tilt-compensated compass method [21–24] (see “Glossary” for a definition of dead-reckoning-related terminology used throughout).

The utility of dead-reckoning depends on the accuracy of speed and heading estimates (see Table 1) and, due to the nature of vector integration, dead-reckoned tracks accumulate errors (commonly termed ‘drift’) over time [15, 25, 26]. As a result, periodic ground-truthing by a secondary source is important for maintaining the accuracy of animal paths with all its underlying scales and tortuosity of movement [9, 10, 27]. For this reason, dead-reckoning data are normally enhanced by combining it with other methods for providing Verified Positions (VPs). These are primarily; direct observation [e.g., 28], light intensity-based geolocation [e.g., 29], VHF—[e.g., 30], acoustic—[e.g., 31] and GPS telemetry [e.g., 26]. Other, less utilised, systems that may also have merit at sites frequented by the tagged animals, include radio frequency identification (RFID) stations [cf. 32], camera traps [cf. 33] and video footage, such as closed-circuit television (CCTV) surveillance [e.g., 34]. Although all these systems are subject to a number of issues that can make their positional fixes temporally widely spaced [e.g., 4, 35, 36], inaccurate [e.g., 37, 38, 39] or impossible [e.g., 40, 41, 42], they can be critical in providing ground-truthed positions, even infrequently, with which to reset accumulated drift [9, 26].

Of the above VP options, GPS-corrected dead-reckoning is the most widely used and there is a marked bias towards marine studies [e.g., 10–12, 15–17, 19, 27, 43, 44–56]. This is likely for logistical reasons, with many aquatic animals being larger (and thus can carry larger/more devices) than their terrestrial counterparts [57],

whilst the utility of transmission telemetry is restricted to periodic resurfacing events [58]. Moreover, speed can be more easily measured or approximated in water, with previous studies obtaining estimates via acoustic flow noise [e.g., 59], passive sonar [e.g., 60], pitch and change in depth [e.g., 11] and speed sensors [cf. 16, 61, 62]. The efficacy of such techniques diminishes within the aerial environment, principally, due to the marked difference between water and air density [cf. 63] and the current speed and volatility of wind [cf. 64, 65]. Indeed, this may explain why, in part (to our knowledge), only one study to date has dead-reckoned a volant species [66]. More recently, dynamic body acceleration (DBA, see Wilson *et al.* [67], for recent review) has been validated as a proxy of speed for terrestrial animals [68, 69] although there are still very few studies that use the dead-reckoning method in terrestrial animals [e.g., 9, 10, 26, 34, 70, 71].

We suggest that another reason that Verified Position Correction (VPC) dead-reckoning has been little used relates to the apparent difficulty and poor accessibility of the analytical processes involved. With this in mind, the primary aim of this paper is to provide potential users with a clear, concise roadmap for implementing dead-reckoning protocols. Specifically, we revisit the dead-reckoning methodology, from calibrating magnetometry data and deriving heading (tilt-compensated compass method), to VPC dead-reckoning within both terrestrial and fluid media. We provide simplistic conceptual explanations and mathematical protocols and describe the pitfalls within the procedure that can increase error. We also translate the relevant equations into complementary R code [cf. 93, available at 94] throughout the text, with fully annotated scripts deposited in Additional files 2, 3, 4, 5 and GitHub [available at 95].

In addition to the above, we outline recent advances to the VPC dead-reckoning technique. For use in terrestrial environments, this includes implementing step counts as a distance measure, by-passing dynamic body acceleration (DBA) as a proxy of speed, and assessing the value of ‘reverse dead-reckoning’ (useful when VPs are concentrated to the latter end of an animal’s trajectory). For marine and aerial environments, we demonstrate the value of integrating tidal-/air current data with dead-reckoned vectors (hereafter termed ‘current integration’) to reduce errors attributed to drift [cf. 46, 92]. Across all three media of travel (land, water and air), we show the value of incorporating different speed coefficients according to behaviour types. In addition, we provide examples of the various methods by which VP data can be under-sampled (relevant for high-res GPS datasets) prior to correcting dead-reckoned tracks and discuss the scales at which users should consider VP correction (which depend on the details of species-specific movement and

Table 1 Possible system errors that can affect the utility of animal dead-reckoning within the 'tilt-compensated compass' framework

System error	Reasons for error	Underlying causes	References	Possible mitigation measures
Derived heading	Errors in deriving static acceleration (postural) estimates	During bouts of high centripetal (turning) acceleration Free-falling behaviour	[65, 72–75]	Gyro-integrated data [cf. 76, 77]
	Using Euler angles (angle of rotation about each axis of a given coordinate system)	The orientation of the device with respect to the earth's frame of reference (cf. Additional file 1: Text S2) can only be defined reasonably at angles less than perpendicular or less than a 180° inversion (dependent on pitch and roll equations used—cf. "VPC dead-reckoning procedure in R" section) from their longitudinal and lateral axes of 'normal' posture (otherwise, unstable measures arise from the Gimbal lock singularity complex [cf. 78], whilst x, y and z values can become inverted and/or represent different 'surge', 'sway' and 'heave' planes)	[21, 79–82]	Quaternion-estimated heading [cf. 83, 84, 85]
	Tag placement/dislodgment	In line with the above—range for accurate angular (pitch and roll) measures are restricted in one or more dimensions Heading will be biased according to the degree of displacement about the z-axis	[19, 86]	Ensure tag orientation is noted during deployment and retrieval operations (and subsequently used in corrections)
	Variations in the strength and declination of magnetic fields	Animals that undertake long journeys (regionally/ globally) Environmental and man-made magnetic noise (iron distortions)	[87]	Ensure at least one magnetic calibration procedure is carried out (see Additional file 1: Text S3 for details) and apply magnetic declination offset to heading values where required
Derived speed	Deviations of the DBA ~ speed relationship	Load bearing Moving over a deformable substrate/changeable incline Changing gait Moving within fluid media Gliding/thermalling behaviour	[12, 65, 68, 69, 88–91]	Iteratively modulate the gradient and/or intercept within the DBA ~ speed linear regression according to environmental circumstance and mode of movement [cf. 68] By-pass DBA (e.g., use speed/acoustic sensors, step/tail-/wing beat frequency, vertical speed, etc. [e.g., 47])
Both	External forces (e.g., current vectors in air- and water flow)	Decreases the signal-to-noise ratio of motion sensor data Affects the relationship between an animal's (longitudinal axis) direction of travel from their true vector of travel Some animals do not always move in the same direction as their anterior–posterior axis	[11, 15, 46, 92]	Smooth postural (and pre-derivative data)/DBA estimates [cf. 67] Incorporate current flow vectors within the dead-reckoning procedure

length of data acquisition). We specifically demonstrate the above using our R-functions (*Gundog.Tracks* being the primary function for dead-reckoning), providing examples of its utility across various scenarios. Lastly, we highlight the relevance of heading and distance correction factors (derived from the VPC procedure), which can also be used to interrogate the animal–environment interaction and biases stemming from animal tag performance.

To illustrate our approach, we use three model species (the African lion *Panthera leo*, the Magellanic penguin *Spheniscus magellanicus* and the Imperial cormorant *Leucocarbo atriceps*) that cover almost two orders of size magnitude in body mass and that operate in markedly different environments and at different scales of movement. To make this review more broadly applicable to researchers of varying dead-reckoning and R knowledge, we have departed from a traditional article format and instead, split this body of work into two distinct sections: firstly, we provide an overview of the critical *Gundog.Tracks* function and provide a brief review of the conceptual workflow (“Implementation of *Gundog.Tracks*” section). With respect to this, we discuss the relevant strengths and limitations of the current dead-reckoning framework and the key considerations involved. Secondly, we detail each ‘potential’ stage of the VPC dead-reckoning procedure with exemplar mathematical equations and R syntax (“VPC dead-reckoning procedure in R” section).

Implementation of *Gundog.Tracks*

We expand on key concepts in Additional file 1 and provide complimentary R scripts (outlined below) in Additional files 2, 3, 4 and 5. We also supply an example data set of a Magellanic penguin walking out to sea in Additional file 6, which can be used to trial each of the provided R scripts and perform the full dead-reckoning process. Mathematical equations are referred to as ‘Eqs. 1–33’ and R syntax as ‘R_x’ where ‘x’ is the reference number. To simplify concepts, we use base R syntax (wherever possible) and typically use vectors to demonstrate points made, though ‘df\$’ directly before the variable name indexes data retained within data frame columns (assuming data frame is called ‘df’). We note, however, that more efficient code implementations are possible (e.g., *data.table* [96] and *lapply()*) than presented here, especially for large data, but here wanted to make the code as readable as possible in this manuscript, especially to persons not familiar with complex coding. More efficient code will be implemented through updated GitHub versions of the functions. See Additional file 1: Text S1 for our model species’ device setup and capture protocol and the glossary for a definition of dead-reckoning related terminology.

User functionality

Gundog.Tracks is an all-encompassing dead-reckoning function that can be used to dead-reckon animal paths travelling terrestrially or through fluid media. Table 2 details all the function’s input requirements/options.

Reverse dead-reckoning

Dead-reckoning backwards is useful when the start position is unknown, but the finishing coordinates are known. For example, central-place foraging, diving animals returning to land from the sea may not acquire a satellite fix for an appreciable period of time following submersion in water which can make determining the start position difficult. So, when VPs are skewed to the latter part of the track, it may be beneficial to start the iterative dead-reckoning process from that end. This involves reversing the order of data to be dead-reckoned and changing heading values by 180 degrees prior to dead-reckoning.

Integrating current vectors

Wind or ocean currents can change the relationship between an animal’s (longitudinal axis) bearing and speed of travel from their true vector of travel [46, 92]. This drift can be incorporated within movement paths by advancing each iterated dead-reckoned vector according to the direction and speed of the current at that point in space and time (cf. Fig. 1).

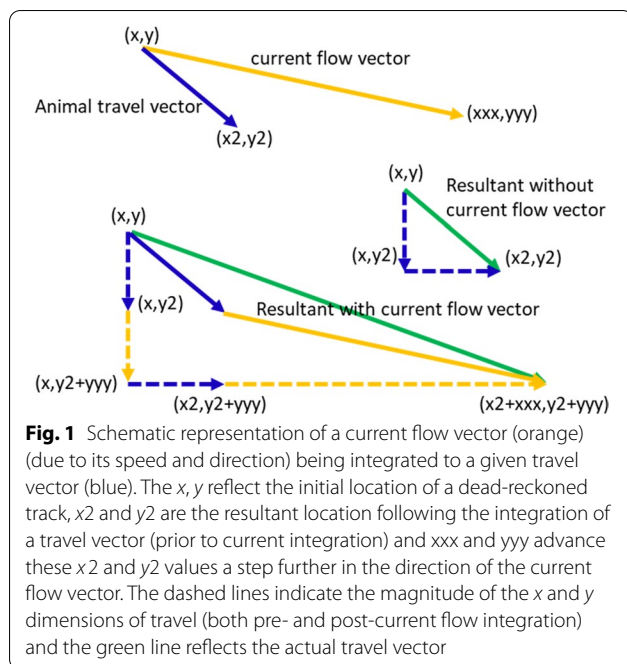
DBA–speed derivation

Given the approximate linear relationship between DBA [sensu 67] and terrestrial animal speed [$\text{speed} = (\text{DBA} \cdot m) + c$], DBA estimates can be multiplied by a gradient, m (the multiplicative coefficient) and summed with an intercept, c (the constant) to derive speed [10, 26]. These values are typically substituted with results from DBA–speed linear regression estimates, such as from treadmill tests or using GPS-derived speed [26, 69, 97, 98]. The m -coefficient should be selected such that (uncorrected) dead-reckoned tracks accord with the apparent straight-line distance between VPs. Importantly, the DBA–speed relationship may be a function of terrain-type (e.g., sand vs. concrete), animal state (e.g., weight variation) and mode of movement (e.g., running vs. climbing) [cf. 68]. For instance, a condor gliding within a thermal would have high speeds, despite having negligible DBA, while an Ibex traversing across different substrate types and gradients would impart varying magnitudes of acceleration that may scale non-linearly with a change in stride gait. It may be of value, therefore, to iteratively change the supplied m (and possibly c) values between VPs according to behaviour and environment. The user

Table 2 Gundog.Tracks input fields and description of their role

Function input	Description	Ref
TS	Timestamp - POSIXct object. No missing data (NA's) permitted	-
h	Heading (0° to 360°) - No missing data (NA's) permitted	-
v	DBA (g or m/s ²) or speed (m/s) - No missing data (NA's) permitted	-
elv	Elevation / depth data (m) - No missing data (NA's) permitted	0
p	Pitch (°) – Only supply if user wants radial distance modulated according to pitch (cf. "VPC dead-reckoning procedure in R" section—Eq. (27)). No missing aadata (NA's) permitted	NULL
cs	Current speed (m/s) - Supplied as a single value or vector/column of changeable values. NA's are replaced with the most recent non-NA prior to it (observations carried forward)	NULL
ch	Current heading (0° to 360°) - Supplied as a single value or vector/column of changeable values. NA's are replaced with the most recent non-NA prior to it (observations carried forward)	NULL
m	Multiplicative coefficient (gradient) - If speed (m/s) supplied for v, then m must be 1. Supplied as a single value or vector/column of changeable values	1
c	Constant (y-intercept) – If speed supplied for v, then c must be 0. Supplied as a single value or vector/column of changeable values	0
ME	Marked Events – 0 denotes periods of stationary behaviour and 1 (or any integer number > 0) denotes periods of traversing movement. ME overrides initial speed input / DBA-derived speed (calculated within the function itself). NA's and character values are replaced with zero.	1
lo	Starting longitude coordinate to advance dead-reckon track from – Decimal format, e.g., 26.31989	0
la	Starting latitude coordinate to advance dead-reckon track from – Decimal format e.g., -06.11995	0
VP.lon	VP longitude coordinates – Decimal format. Missing relocation data expressed as either NA's or 0's. First (or last if reverse dead-reckoning) element/row allocated as lo within the function	NULL
VP.lat	VP latitude coordinates – Decimal format. Missing relocation data expressed as either NA's or 0's. First (or last if reverse dead-reckoning) element/row allocated as la within the function	NULL
VP.ME	TRUE = Supplied VPs removed at times when ME = 0 (relevant for high-res VP datasets, when location error is high during rest). Note, this does not remove the element/row allocated as lo/la	FALSE
method	How the function under-samples VPs prior to correction (subsequent to the VP.ME subset, if set to TRUE) – "divide" = Fix kept every x (thresh) segments of supplied VPs, based on row number. The first and last fixes are always included "time" (s) = Fix kept every x (thresh) accumulated seconds (or the next available fix after a period of missing locational data (≥ thresh). The first and last fixes are always included "distance" = Fix kept every x (thresh) proportional segments of the total accumulated distance (m) between supplied VPs (using the stepping interval 'dist.step'). The first and last fixes are always included "all" = Every supplied VP kept (irrespective of thresh value)	NULL
thresh	Threshold - Degree of VP under-sampling prior to dead-reckon correction. The frequency of under sampling depends on the method selected	1
dist.step	The stepping interval used for calculating distance between VPs, both within the VP summary distance metrics (see Table. 3) and within the 'method = distance' VP under-sampling protocol prior to VPC. For example, dist.step = 5 computes distance between every 5th VP (irrespective of the time difference between them)	1
bound	TRUE = VPC dead-reckoning is bounded by the first and last VP present FALSE = VPC dead-reckoning is unbounded by the last available VP. The last dead-reckoned track segment inherits the previous correction factors	TRUE
Outgoing	TRUE = 'normal' dead-reckoning procedure FALSE = Reverse dead-reckoning. Note la and lo positions should now be the finishing longitude and latitude coordinates, respectively	TRUE
Plot	FALSE = No summary plots TRUE = R graphics window initialized: VPC = 4 summary plots / no VPC = 1 summary plot (cf. top left) Top left) Uncorrected dead-reckoned track (blue) and VP track (red). If currents are supplied, the blue track has currents integrated and an additional green track with no current integration is plotted Top right) VPC dead-reckoned track (blue) in relation to VP track (red) Bottom left) Net error (m) between VPs and dead-reckoned positions (un-corrected = red and corrected = black). If currents are supplied, then uncorrected with no current integration = green and uncorrected with current integration = red. Bottom right) VPC corrected dead-reckoned track	TRUE

Ref. refers to the default value when no input is stated. Red shading represents required user inputs and green and orange shading reflect optional inputs (the latter change when using VPC). Note that if speed estimates (v) are directly inputted into the function then m and c (and possibly ME) defaults should not be changed. If either one of the VP.lon, VP.lat or method inputs is specified as NULL, then no VPC will occur



may also opt to supply a ‘marked events’ (ME) vector (a marked event is a term we use that refers to a number of sequential (in time) data points within a dataset coded by integer values) to ensure dead-reckoned tracks are not advanced with non-animal movement behaviours. Within *Gundog.Tracks*, ME values of one or greater reflect progressive movement, and zero values code for stationary behaviour—dead-reckoned tracks are not advanced when $ME=0$ (irrespective of the allocated speed). For example, in its simplest form, ME could be filled with binary 0 s and 1 s as governed by a DBA threshold (labelling the ME vector 0 in sleep and resting behaviour).

Pre-determining speed

For terrestrial species (specifically bipeds and quadrupeds), the interplay between peak heave acceleration amplitude and periodicity may be a useful indicator for the movement gait adopted [99], which may help decide the m -coefficient in the DBA–speed relationship [69]. There may be times, however, when DBA is an unreliable proxy of terrestrial speed [cf. 68]. At this time, given that the stride cycle can be easily detected by cyclic peaks in a given acceleration channel [e.g., 100, 101, 102], peak periodicity (and amplitude) may be used as a proxy of distance moved by providing a distance per step estimate (assuming constant distance travelled between step gaits if only concerning step periodicity—cf. “VPC dead-reckoning procedure in R” section and Additional file 1: Text S4).

DBA is a weak proxy of speed for many marine animals because overall body tissue density changes with depth when air is associated so that speed may be invariant of the movement kinematics [cf. 103, 104]. DBA is also a weak proxy for flying animals that glide at constant velocity, use thermals or bank [cf. 105]. One of the most common methods for determining animal speed in water is via devices that estimate flow or resistance rate [16, 19, 64, 106]. These often have appreciable limitations, with currents, biofouling, blockage and turbulence affecting performance [64], and many of these issues are applicable to volant species, so that bird speed measures are typically restricted to GPS-derived estimates of ground speed [cf. 107]. In the absence of a reliable motion sensor-derived speed proxy, previous reported approximated speed estimates according to movement modes and/or topological whereabouts can be used [cf. 30]. For example, for various diving animals such as penguins, a simple depth threshold may prove effective to differentiate between various previous reported modal ‘surface-resting’ and ‘underwater-commuting’ speeds [61, 108–110]. For volant species, whilst wingbeat frequency or amplitude does not scale reliably with air speed [cf. 111], the interplay between both can decipher various flight modes (e.g., ‘cruising speed’ vs taking off/landing) [65, 112]. Furthermore, tail beat frequency has been shown to be a good predictor of swimming speed for various fish species [113–115]. For diving animals, a proxy for horizontal speed can be obtained based on animal pitch and rate of change of depth [11, 116]. Specifically, the rate change of depth is divided by the tangent of the body pitch.

In any case, when high-resolution VP data are available (e.g., 0.01–10 Hz GPS), for instance, during short-term trial deployments, speed estimates can be compared alongside those derived between VPs and approximated according to behaviour type (elucidated from, for example, accelerometer—[e.g., 117, 118], magnetometry—[e.g., 105, 119], depth- [e.g., 120] or altitude—[e.g., 65] data), and uncorrected dead-reckoned tracks can be compared alongside VPs to determine where biases may occur visually. Furthermore, the correction factors obtained from the VPC process are viable comparators for detecting consistent under- or over-estimations of speed and/or heading offsets (e.g., due to tag placement). Essentially, when empirical speed evidence is unavailable, the user can ad hoc iteratively adjust allocated speed values or the underlying DBA–speed coefficients until uncorrected dead-reckoned track segments scale proportionately to their aligned ground-truthed positions (pre-VPC). Within *Gundog.Tracks*, the user can modulate m , c and ME values to switch between predetermined speed ($m=1, c=0$), DBA-derived speed ($m>0, c\geq 0$) and stationary behaviour ($ME=0$).

VPC procedure

Ground-truthing dead-reckoned tracks typically involves the linear drift correction method [cf. 26, 46], outlined in Constandache et al. [121] and Symington and Trigoni [122]. In essence, a shift vector aligns the starting dead-reckoned path segment with the VP at time point one, after which the difference between the VP and dead-reckoned path segment at time point two is calculated to provide a correction vector that is applied linearly between time point one and time point two. Our method follows the protocols outlined by Walker et al. [9], whereby the underlying correction coefficients (hereafter termed ‘factors’) for both heading and (radial) distance are calculated—adjusting the length and heading at each dead-reckoned path segment until the end points align to each VP along the path. This process requires the trigonometric ‘as the crow flies’ Haversine formulae [123–125] which allows one to translate a distance across the curvature of the Earth’s surface (detailed within “VPC dead-reckoning procedure in R” section). The advantage of this method is that, whilst correction factors are constant between VPs, it does not assume that the dead-reckoned path deviates linearly over time from the true path because (radial) distance is multiplied by the distance correction factor. This ensures that parts of track where the animal is determined to be stationary (e.g., ME = 0) are left unaltered. The function’s method of VPC, automatically handles NaN and Infinite (Inf) values which can arise during the derivation of the distance correction factors (when no dead-reckoned movement occurs between successive VPs—detailed within “VPC dead-reckoning procedure in R” section). It is worth noting that even animals that travel in 3-D can be subject to the 2-D dead-reckoning formulae and Haversine computation of distance correction factors because we typically assume that both dead-reckoned- and VP positions are aligned in vertical space (assuming reliable pressure—[60]/depth [13] data) and attempt to control for the horizontal component of speed [e.g., “VPC dead-reckoning procedure in R” section—Eqs. (25, 27)] pre-correction. Although not covered here, we acknowledge that various state–space modelling techniques have also been developed to georeference dead-reckoned tracks [e.g., 11, 47].

Default inputs for calculations and outputs

Gundog.Tracks default input takes the form:

```
Gundog.Tracks(TS, h, v, elv=0, p=NULL,
cs=NULL, ch=NULL, m=1, c=0, ME=1,
lo=0, la=0, VP.lon=NULL, VP.lat=NULL,
VP.ME=FALSE, method=NULL, thresh=1,
dist.step=1, bound=TRUE, Outgoing=TRUE, plot=FALSE),
```

with input modulated according to the animal in question and data available (see Fig. 2).

The function outputs a data frame containing various descriptive columns which, depending on the input, includes (but is not limited to):

- The correction factors used
- Heading and radial distance estimates (both pre- and post-current integration and/or VPC)
- Distance moved and speed estimates (both in 2-D and 3-D when elevation/depth data supplied)
- Net error between dead-reckoned positions and VPs (both pre- and post-correction)
- Various VP summaries including notation of when VPs are present and which fixes were used in the correction process.

When specified, 2-D summary plots demonstrating the relationship between dead-reckoned positions and VPs (both pre- and post-current integration and/or VPC) are provided (e.g., Fig. 3). Table 3 details all the function’s available outputs (modulated according to input). *Gundog.Tracks* uses the *na.locf()* function from the ‘*zoo*’ package [126] and the *slice()* function from the ‘*dplyr*’ package [127] (both are checked as dependencies and installed when required within this function). Output 2-D distance/speed estimates are calculated with the Haversine formula. When depth/elevation data are supplied (and changes between sets of coordinates) 3-D distance/speed estimates are calculated with a variant of the Euclidean Formula—converting *x*, *y*, *z* from polar to Cartesian coordinates, and incorporating the Earth’s oblate spheroid [cf. World Geodetic System (WGS84)], via conversion from Geodetic- to Geocentric-latitude [cf. 128].

The interplay between numerical precision in R, correction rate and net error can make more than one round of adjustment necessary for dead-reckoning fixes to accord exactly with ground-truthed locations (cf. Fig. 4a), particularly given that slight discrepancies accumulate over time. Each iteration of the correction process produces a tighter adherence between estimated and ground-truthed positions [cf. 9]. Typically, this does not involve more than two rounds of VPC to achieve a maximum net error of 0.01 m (the threshold used within *Gundog.Tracks*) across a *ca.* (1 Hz) 2-week-long track. For an indication of processing times see Additional file 1: Text S6, Fig. S4; for example dead-reckoning a lion at 1 Hz for 7 (continuous) days (with *plot*=TRUE, *dist.step*=5, *VP.ME*=TRUE, *method*="time" and *thresh*=3600) took 25 s to compute (on a MSI GP72 7RD Leopard laptop with intel core i7 processor). Logically, the net error between VPs and (corrected) dead-reckoned positions is positively correlated to the time between corrections (cf.

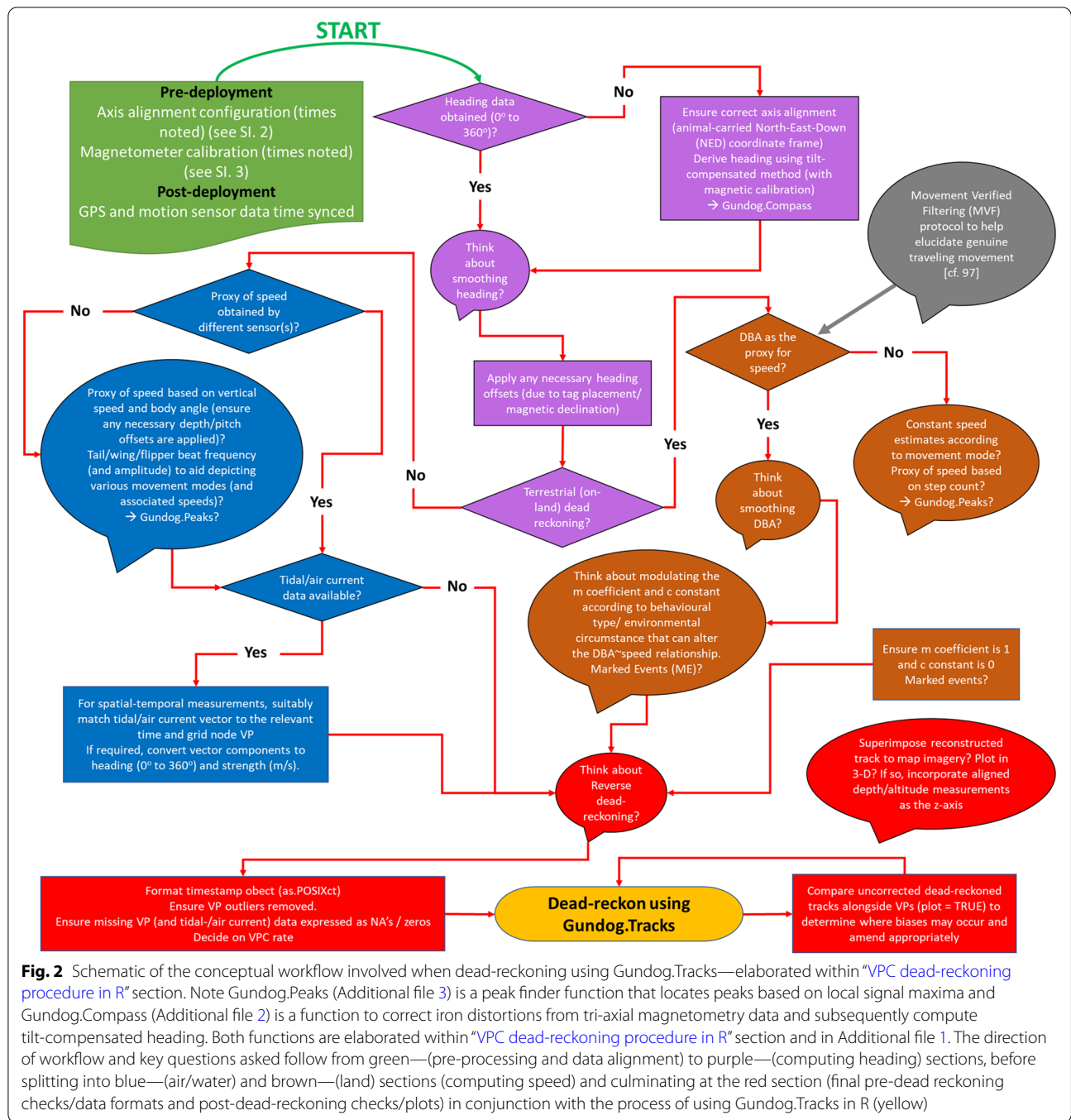
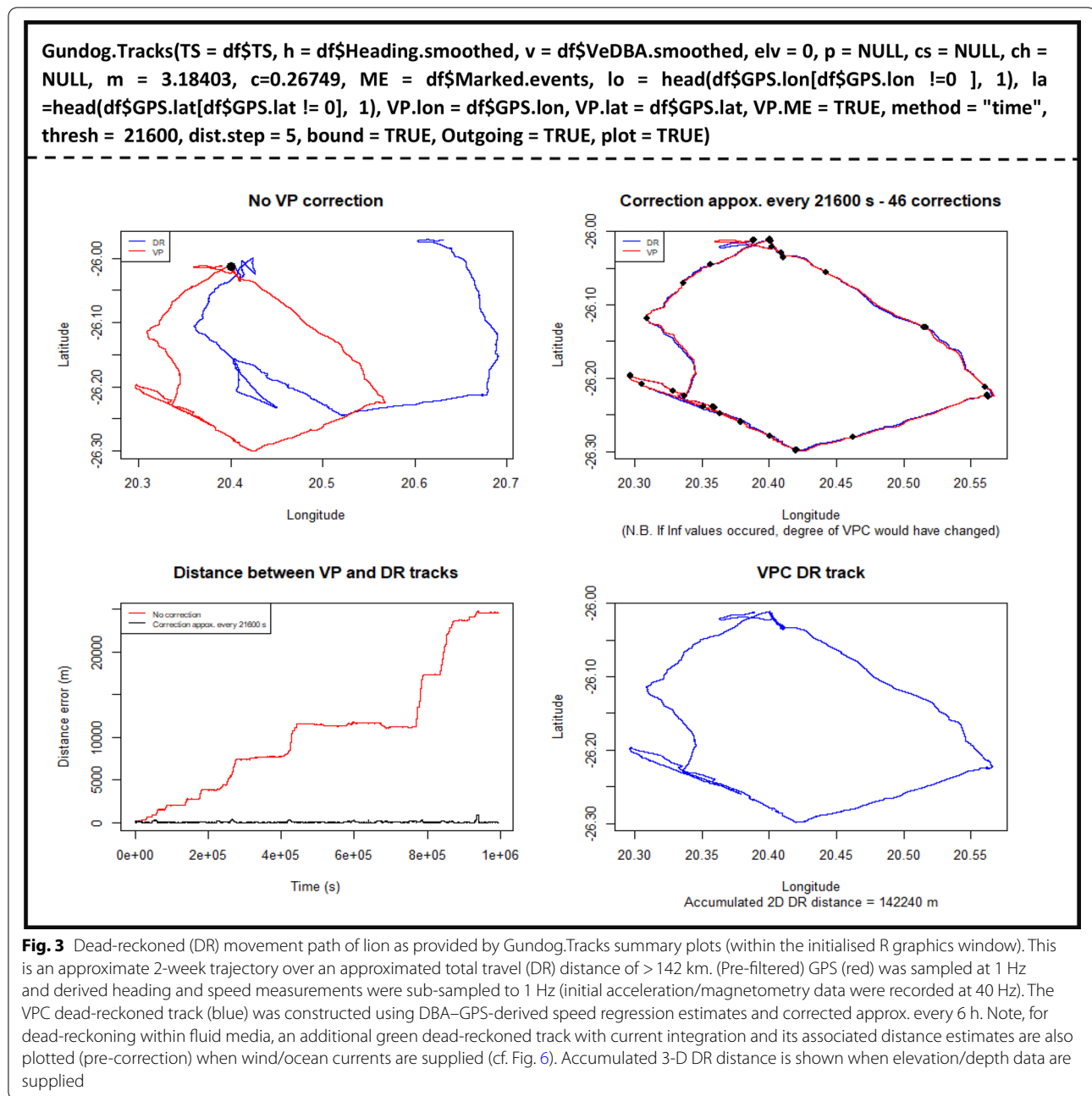


Fig. 2 Schematic of the conceptual workflow involved when dead-reckoning using Gundog.Tracks—elaborated within “VPC dead-reckoning procedure in R” section. Note Gundog.Peaks (Additional file 3) is a peak finder function that locates peaks based on local signal maxima and Gundog.Compass (Additional file 2) is a function to correct iron distortions from tri-axial magnetometry data and subsequently compute tilt-compensated heading. Both functions are elaborated within “VPC dead-reckoning procedure in R” section and in Additional file 1. The direction of workflow and key questions asked follow from green—(pre-processing and data alignment) to purple—(computing heading) sections, before splitting into blue—(air/water) and brown—(land) sections (computing speed) and culminating at the red section (final pre-dead reckoning checks/data formats and post-dead-reckoning checks/plots) in conjunction with the process of using Gundog.Tracks in R (yellow)

Fig. 4b) [cf. 46], although the rate of net error ‘drop-off’ is dependent on the accuracy of the initial (uncorrected) dead-reckoned track (cf. Fig. 5), itself, modulated by the extent of system errors (Table 1) and initial user-defined track scaling.

Within this process, people assume VPs to be perfect, however, across all VP determining methods, the rate and accuracy of data acquisition is highly moderated

according to the permissiveness of the environment, such as high-density shrub or submersion in water [e.g., 38, 129, 130]. GPS technology is arguably the most popular and widely used method for determining estimates of free-ranging animal movement [cf. 131, 132, 133]. This is because inspection of data is less complex and time-consuming than some of the alternatives, whilst improvements in design and battery longevity have enabled



GPS units to be attached to a plethora of animals (up to almost four orders of magnitude in size and mass [cf. 11, 134]) and record at high frequencies (e.g., ≥ 1 Hz [131, 135]). Consequently, GPS units are unparalleled for providing such detailed quantification of space-use outside of the VPC dead-reckoning framework, and are the most utilised VPC method within (including the case study datasets within this study). However, locational accuracy (excepting precision error radius [cf. 136] and variable latency [cf. 137]) can vary by a few metres or be

appreciably more depending upon the propagation of signal quality and/or receiver reception capability [38, 138, 139]. As such, VP error becomes more relevant at smaller scales of assessed movement and this is the reason why VP distance-moved estimates can go from being typically underestimated at low frequencies (due to linear interpolation of tortuous movements) [26, 140, 141] to overestimated at high frequencies [97, 136] and result in highly variable correction factors within the VPC dead-reckoning process [cf. 10]. Indeed, judicious selection of VPC

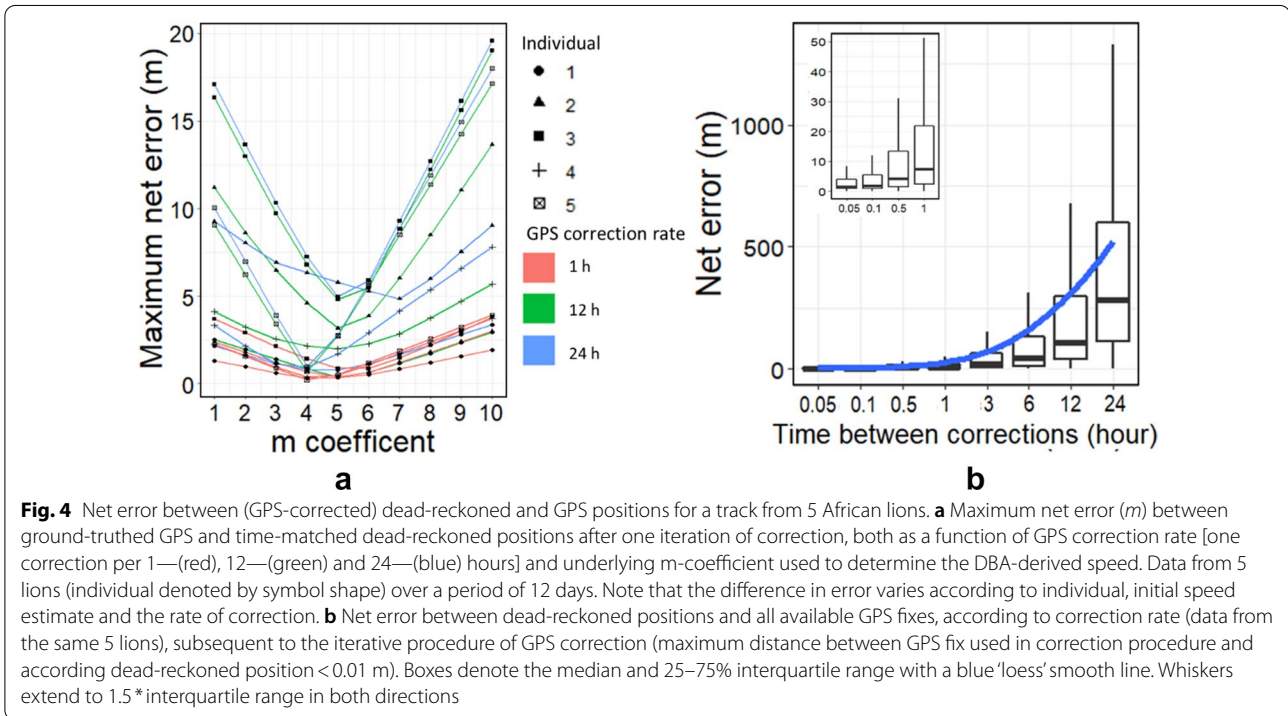
Table 3 Gundog.Tracks data frame output names and their parameters

Function output	Description	Ref
Row.number	Row number	
Timestamp	Supplied timestamp - POSIXct object	
DR.seconds	Accumulated time (s) based on the supplied timestamp	
Heading	Supplied heading (0° to 360°)	
Marked.events	Supplied Marked events (or replicated default)	
DBA.or.speed	Supplied DBA (g or m/s^2) or speed (m/s)	
Pitch	Supplied pitch (°)	
Radial.distance	The calculated q-coefficient (prior to VPC) (see "VPC dead-reckoning procedure in R" section—Eq. (26))	#
Elevation	Supplied elevation / depth (m)	
Elevation.diff	Rate change of supplied elevation/depth (m/s) - (elevation difference / time difference between rows)	
Current.strength	Supplied current speed (m/s)	
Current.heading	Supplied current heading (0° to 360°)	
Heading.current.integrated	Updated heading (0° to 360°) following addition of current vectors (prior to VPC)	#
Radial.distance.current.integrated	Updated q-coefficient following addition of current vectors (prior to VPC)	#
DR.longitude	Dead-reckoned longitude coordinates – Decimal format (prior to VPC)	
DR.latitude	Dead-reckoned latitude coordinates – Decimal format (prior to VPC)	
DR.longitude.corr	Corrected dead-reckoned longitude coordinates – Decimal format (post VPC)	
DR.latitude.corr	Corrected dead-reckoned latitude coordinates – Decimal format (post VPC)	
Dist.corr.factor	Distance correction factor (observations carried forward)	↑ #
Head.corr.factor	Heading correction factor (0° to 360°) (observations carried forward)	↑ #
Heading.corr	Corrected heading (0° to 360°) (post VPC)	#
Radial.distance.corr	Corrected q-coefficient (post VPC)	#
Distance.error.before.correction	Distance (m) between uncorrected dead-reckoned positions and VPs (observations carried forward), subsequent to sub-sampling according to ME, if VP.ME = TRUE	↑
Distance.error.after.correction	Distance (m) between corrected dead-reckoned positions and VPs (observations carried forward), subsequent to sub-sampling according to ME, if VP.ME = TRUE	↑
DR.distance.2D	Two-dimensional distance moved (m) between dead-reckoned fixes	*
DR.distance.3D	Three-dimensional distance moved (m) between dead-reckoned fixes	*
DR.cumulative.distance.2D	Accumulated two-dimensional distance moved (m) between dead-reckoned fixes	*
DR.cumulative.distance.3D	Accumulated three-dimensional distance moved (m) between dead-reckoned fixes	*
DR.distance.from.start.2D	Two-dimensional (straight-line) distance moved (m) from starting position	*
DR.distance.from.start.3D	Three-dimensional (straight-line) distance moved (m) from the starting position	*
DR.speed.2D	Horizontal speed (m/s) (DR.distance.2D / time difference between rows)	*
DR.speed.3D	Total speed (m/s) (DR.distance.3D / time difference between rows)	*
VP.seconds	Accumulated time (s) between supplied VPs (observations carried forward)	
VP.longitude	Supplied VP longitude values (observations carried forward), sub-sampled according to ME, if VP.ME = TRUE	↑
VP.latitude	Supplied VP latitude values (observations carried forward), sub-sampled according to ME, if VP.ME = TRUE	↑

Table 3 (continued)

VP.fix.present	Denotes when a fix was present (1) or absent (0), subsequent to sub-sampling according to ME, if VP.ME = TRUE	
VP.used.to.correct	Denotes which VPs were used to correct (1) and which VPs were ignored (0)	
Number.of.VPCs	Increments by 1 each time a VP was used to correct (observations carried forward)	↑
VP.thresh	Replicates the thresh value set (or default) or warns the user that additional VP under-sampling was required if 'Inf' values produced	
VP.distance.2D	Two-dimensional distance moved (m) between VPs, subsequent to sub-sampling according to ME, if VP.ME = TRUE and using the stepping interval 'dist.step'	
VP.cumulative.distance.2D	Accumulated two-dimensional distance moved (m) between VPs, subsequent to sub-sampling according to ME, if VP.ME = TRUE and using the stepping interval 'dist.step'	

The shading of Ref refers to when the outputs occur; red shading = always, purple shading = when pitch data are supplied, green shading = when elevation/depth data are supplied, blue = when current data are supplied and orange = when the user opts to undertake VPC. The symbol * demonstrates that the metrics will be derived from VPC tracks when correction is initialised. Note that, subsequent to reverse dead-reckoning, the data frame is reverted (back to original time order), though as a result, observations appear to be carried backwards in some instances, indicated by ↑. Due to the nature of reverse dead-reckoning (cf. Additional file 4), some input fields are shifted forward one row following the initial inversion of data. As such, fields indicated by #, are one row further forward in time (this is important when relating Head.corr.factor and Heading.corr output to the equivalent (uncorrected) Heading output. However, when currents are integrated, the Head.corr.factor and Heading.corr outputs refer to Heading.current.integrated and these are synchronised row-wise. All heading related data are rotated back 180 degrees following reverse dead-reckoning



rate is critical in maximising dead-reckoned track accuracy when relocation data are taken at fine spatial- and temporal resolutions [26] (cf. Table 2—'VP.ME', 'method', 'thresh' and 'dist.step' inputs to aid in modulating VPC rate). Likewise, the initial screening for location anomalies, across all VP methods and sampling intervals, is important so as to prevent incorrect distortion of tracks. Put simply, the higher the quality of VP data input, the greater the robustness of the VPC dead-reckoning output.

It was suggested by Bidder et al. [10], that the next stage in this work is to derive a standardised set of rules to maximise the value of both GPS (though this applies to any VP method) and dead-reckoned data in line with the questions being asked. We argue that consistent trends in the magnitude and/or bias of correction factors can be used as a diagnostic tool for elucidating: (i) VP inaccuracy (e.g., possibly manifested by extremely high distance and heading correction factors), (ii) required alterations to the DBA-speed relationship [e.g., due to traversing

across different substrates (e.g., Fig. 5)] and (iii) drift due to current vectors [cf. 16, 46] (e.g., Fig. 6).

The case-studies

An important question to address is how often to do VP correction. This is obviously dependent upon the scales of movement elicited and the medium in/on which the animal in question navigates. Put simply, one should VP correct as little as possible, but as much as is necessary and we elaborate on this using our model species operating in different media. Within Fig. 5, the 1 Hz GPS track (blue) is plotted alongside two different dead-reckoned tracks; [(a)=uncorrected and (b)=corrected approx. every 30 min (method="time")] from 12 days of data acquisition of one lion. There were two variations in the method of scaling the dead-reckoned tracks; a track based on a Vectorial Dynamic Body Acceleration (VeDBA) threshold (red), and a track advanced based on periods of identified movement (purple). The *m*-coefficient and *c*-constant values were determined from the VeDBA–GPS speed relationship (Fig. 5, inset a₁) and the Movement Verified Filtering (MVF) protocol outlined by Gunner et al. [97] was used to depict movement and anomalous GPS fixes (green) and to compute reasonable GPS-derived speed estimates. This case study demonstrates three important points. Firstly, on its own, dead-reckoning is subject to substantial drift and so VPC is essential for resetting this error. The more frequent a user corrects, the more accurate the dead-reckon track becomes (relative to VPs), though VP error can also be substantial, especially during rest behaviour (see Gunner et al. [97] for demonstration of this). For collared animals, heading measurements can become inaccurate at times of erratic collar roll (cf. Table 1) and conjointly, GPS performance is also reduced when antenna position becomes compromised [e.g., 142].

Secondly, and in conjunction to the above, irrespective of VPC rate, the initial allocation of speed is important. Here, only dead-reckoning identified movement periods resulted in greater accuracy than just advancing tracks based on a VeDBA threshold. This is because even stationary behaviours can impart appreciable DBA [e.g., 143] (beyond the threshold), and thus wrongly advance tracks. The false patterns of tortuosity created from this, whilst scaled and possibly rotated with VPC (cf. “VPC dead-reckoning procedure in R” section), remain

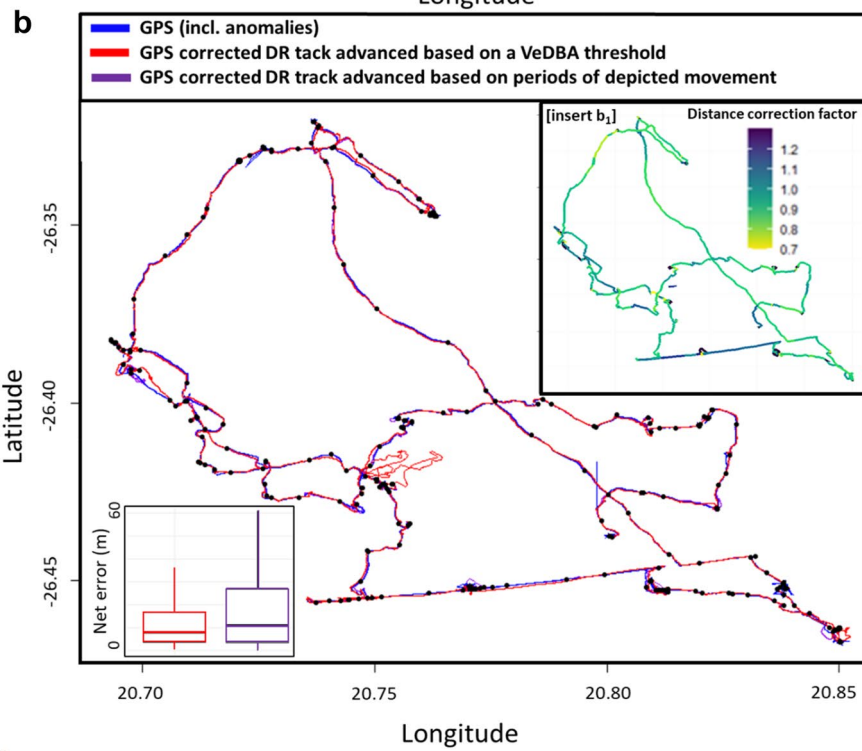
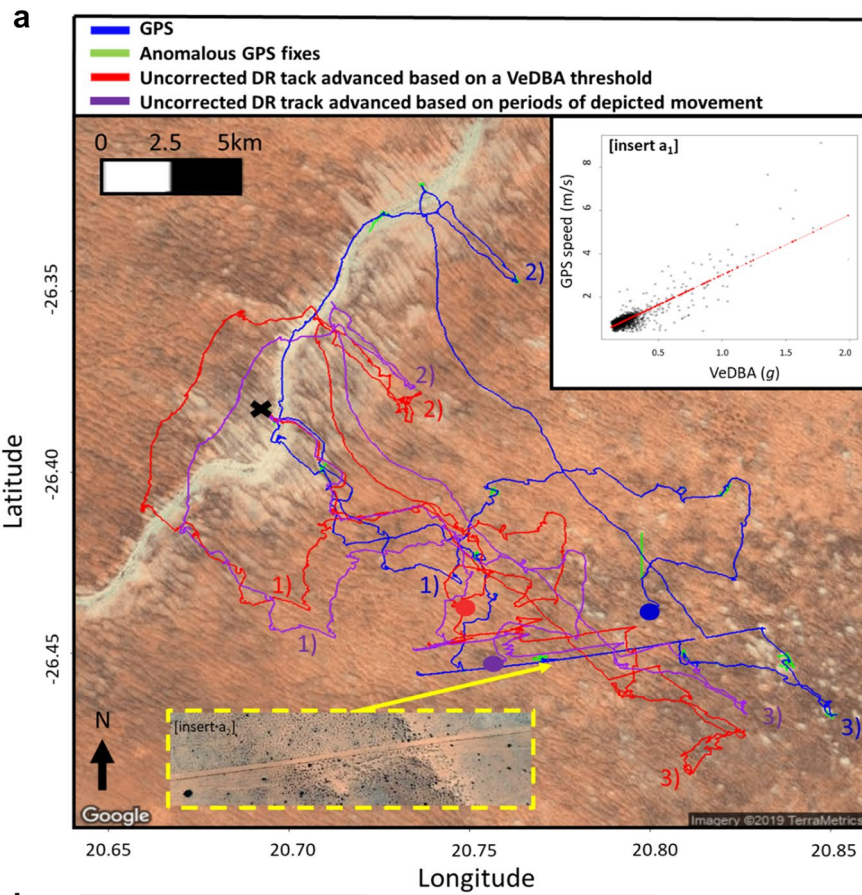
incorporated to some degree. Whilst not illustrated here, advancing tracks without a VeDBA threshold would incur greater error still. Lastly, in this section, the distance correction factor was consistently high (Fig. 5, inset b₁) as the lion travelled along the Botswana fence boundary, perhaps as a result of the animal walking on the compact dirt road at this location (Fig. 5, inset a₂), altering the VeDBA–speed relationship. Such patterns in correction factors (whether consistent or highly variable) can highlight issues with the underlying track scaling.

Where animals move in water or air, obtaining accurate estimates of speed is more difficult without the use of speed sensors. Naturally, the resolution and accuracy of initial dead-reckoning track scaling (pre-VPC) reduces when speed has to be approximated using constant values according to behaviour type (a strategy used here). There is a balance between initial dead-reckoning accuracy and required VPC. The lower the initial track accuracy, the more frequent it should be corrected, and additional drift caused by external-force vectors compounds this issue. Within Fig. 6, we illustrate the value that current correction, dependent on current information, brings to the VPC procedure if the derived track is to be superimposed on the environment. Here, one Magellanic penguin was dead-reckoned with and without tidal vector integration (instantaneous tidal currents were deduced from a 3-D numerical model validated in the region [144], at hourly, 1 km² grid nodes). Commuting speed was allocated 2.1 m/s [cf. 61, 145] and changed according to “VPC dead-reckoning procedure in R” section—R₄₁. Surface period ‘rest’ speed were allocated 0.416 m/s [cf. 108]. VP accuracy improved considerably both pre- and post- VPC when currents were integrated which points to the value of acquiring current data if possible, particularly if VPs are sparse. Notably the combination of dead-reckoning and VP estimation of both movements relative to the ground and fluid, may detail specific orientation strategies used and thus can have value for assessing the ability of drift compensation in aquatic or volant animals [46, 92]

For all our case study animals, GPS units were set to record at 1 Hz. With this temporal resolution (which is not always possible anyway due to the high-power requirements of the GPS), the value of dead-reckoning would seem questionable. However, dead-reckoning can:

(See figure on next page.)

Fig. 5 Dead-reckoned lion track in relation to GPS positions [a uncorrected and b corrected—approx. every 30 min (black circles)]. The start of the track (lo and la) is denoted with a black x. Three corresponding sections of each track are denoted with the same number and the finishing positions denoted with a circle (coloured according to its reference track). Note that the horizontal straight-line sections (cf. yellow arrow) result from the lion following the Botswana boundary fence (which this individual eventually crossed). Mean net error between (corrected) dead-reckoned positions and all available GPS fixes was higher for tracks resolved using a VeDBA threshold (0.11 g), than for tracks advanced only at times of depicted movement using the MVF method



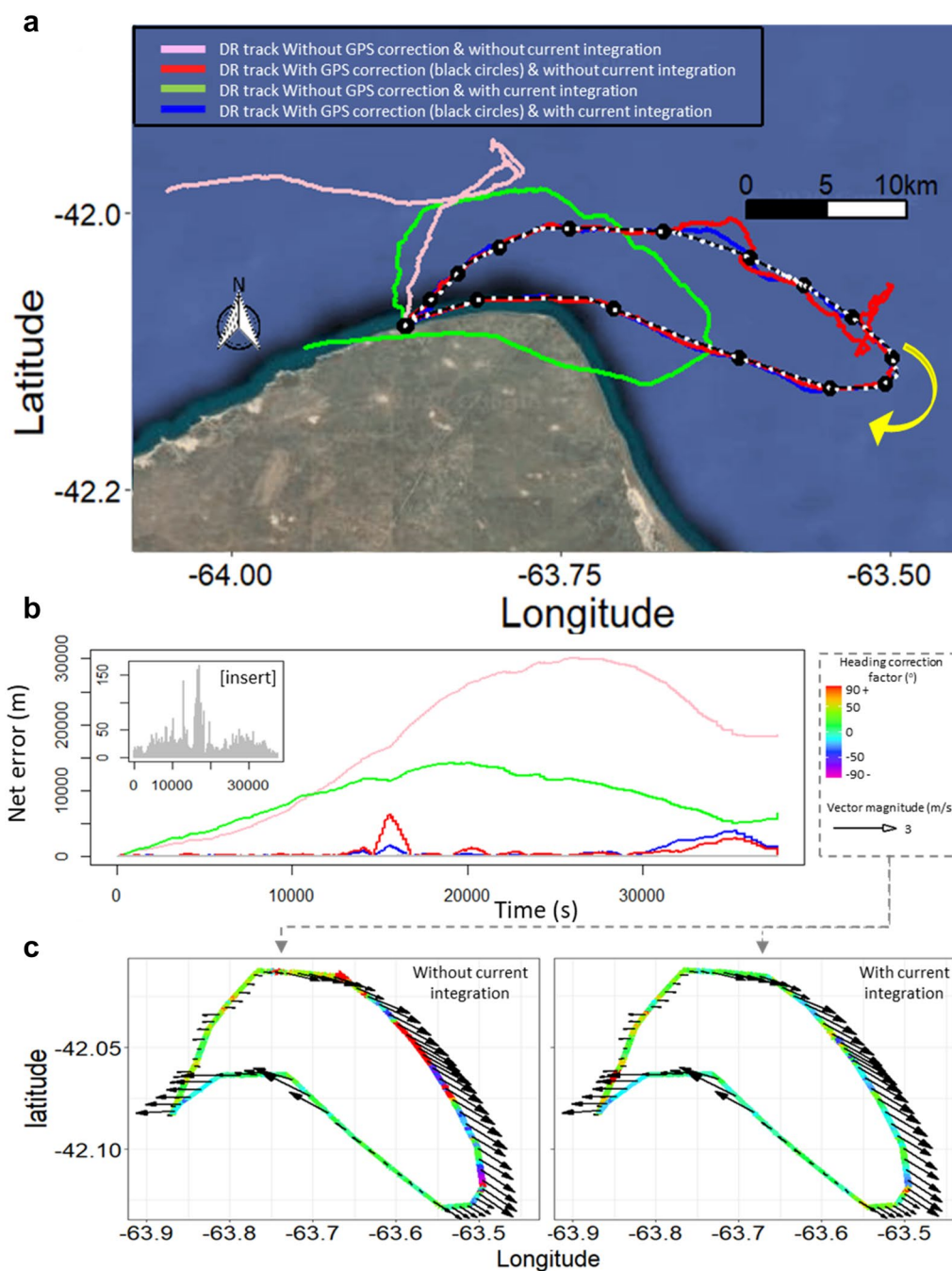


Fig. 6 One Magellanic penguin's dead-reckoned foraging trip at sea, lasting approximately 9 h (yellow arrow denotes the trajectory direction over time. Black track = GPS. Fifteen corrections (black circles) were made (method = "divide"). For comparison, the grey dotted track is the GPS-corrected dead-reckoned track with current integration approx. every 1 min (where possible - method = "time") (a). Note the difference of net error between dead-reckoned positions and all available GPS fixes across the various tracks [insert = grey track] (b). Both uncorrected and corrected dead-reckoned tracks had less error after current integration (black arrows vector every 5 min) and this was reflected in the direction and magnitude of heading correction factors required per unit time (c). Heading correction factors obtained from the track corrected approx. every 1 min; the colour of the scale bar indicates the extent of the heading correction factor required)

(i) work when GPS cannot—such as when an animal is underwater [e.g., 18] or in thick forest [cf. 2] and it can (ii) by-pass the issues arising from GPS inaccuracies such as ‘jitter’ [cf. 97], allowing for more accurate and finer scale delineations of movement. This is illustrated in Fig. 7, in which 12 outgoing (green) and incoming (blue) dead-reckoned trajectories from Magellanic penguins walking to and from their nest are plotted. Incoming tracks were reverse-dead-reckoned (Outgoing=FALSE, bound=FALSE), because the GPS did not always register fixes for minutes after birds left the water and because nest coordinates were known (Fig. 7, inset). This explains why the blue tracks extend into the sea rather than encroach further inland when speed was overestimated. What is evident is that even ‘accurate’ GPS paths are coarsely resolved due to precision errors. Indeed, even with little or no GPS error, this can greatly compromise movement estimates [cf. 136]. Conversely, the precision of the dead-reckoned tracks is only limited by the amount of initial motion sensor data under-sampling (usually required in some capacity to make datasets more manageable and less computationally expensive). Such fine-scale estimates can therefore (with suitable VPC) allow users to define movement in space with unprecedented resolution. The benefit of this is that such resolution can resolve important metrics of movement, such as step duration [cf. 146] and the number and extent of turns made [cf. 147]; useful parameters for investigating navigation and foraging strategies according to environmental circumstance—though, such parameters are also useful without superimposing on the environment. Moreover, even dead-reckoned tracks that are sparsely corrected or never corrected can detail important movement-specific behaviours [12], for example, circling behaviour [148].

Ultimately, the higher the frequency at which dead-reckoning is undertaken, the better the resolution and detail of reconstructed tracks. However, accuracy only improves up to a point because extrapolated travel vectors (heading and speed estimates) nearly always comprise some degree of error (no matter how small) and so, with very high frequencies (>1 Hz), more error is accumulated per unit time [cf. 16, 44]. In particular, when the temporal resolution of dead-reckoning results in a spatial resolution dominated more by sensor noise than by ‘actual’ movement of the animal in question, dead-reckoning accuracy will begin to decrease (at least pre-VPC). The extent of this will depend on the size, speed and lifestyle of the animal in question. For example, the benefits of dead-reckoning a lion at 40 Hz rather than 1 Hz are questionable (how often does a lion turn substantially within a second?), particularly given the additional computation time (cf. Additional file 1: Text S6) and possible

error (relative to VPs). As such, and akin with VP under-sampling, choice of under-sampling data to be dead-reckoned may have implications to the resultant accuracy, and this will be moderated according to the scales (and media) of movement elicited by the animal in question. Beyond this, Fig. 7 also demonstrates the importance of initial track advancement, with three variants used, including step counts instead of DBA.

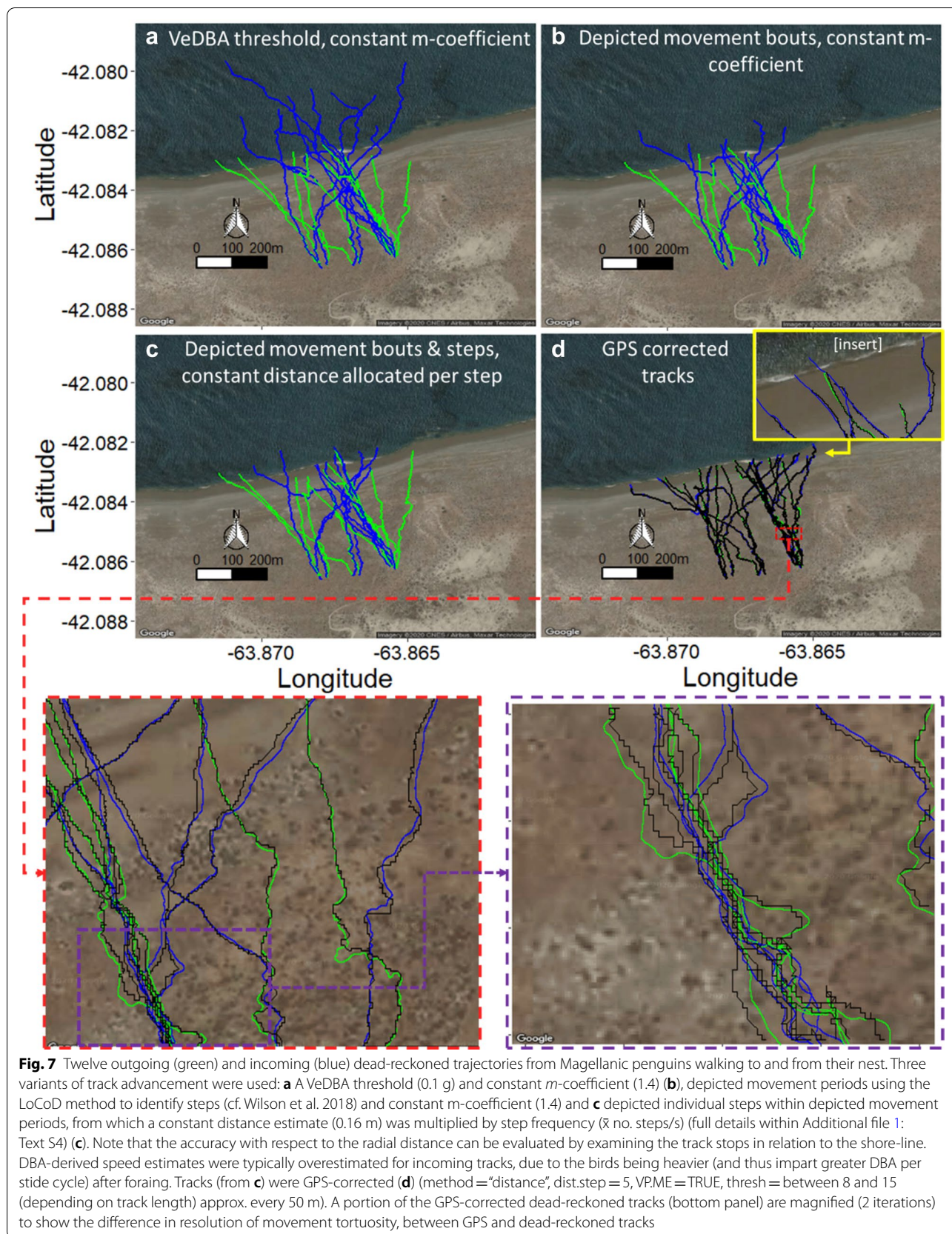
Finally, obtaining accurate estimates of altitude or depth allow users to plot and investigate scales of continuous movement in three dimensions and at times when VP success rate fails completely (such as underwater). We demonstrate this using the Imperial cormorant in Fig. 8. After visual inspection of data, uncorrected tracks were scaled according to the following speeds: periods of flying allocated 12 m/s, surface ‘rest’ periods allocated 0.1 m/s, bottom phase of dives allocated 0.4 m/s and descent and ascent speeds modulated according to “VPC dead-reckoning procedure in R” section—Eq. (25). Note that elevation was not resolved during flying periods (although flying periods were dead-reckoned). Regardless of the current limitations, the VPC dead-reckoning procedure represents a substantial advance for resolving, and thereby allowing investigation of, continuous, fine-scale, free-ranging 2- or 3-D space-use with all its underlying scales of tortuosity and distances moved (e.g., Figs. 7 and 8).

VPC dead-reckoning procedure in R

Preparing the three axes of rotation for derivation of heading

The tilt-compensated compass method is a well-known practice for deriving heading [e.g., 21, 22, 81]. Correct coordinate system axis alignment and suitable calibration of tri-axial magnetometry data [cf. 149] are crucial pre-processors, without which, heading estimates would likely incorporate substantial error [cf. 21, 149]. The tilt-compensated compass method described below (following the framework outlined by Pedley [21]), requires the aerospace (x -North, y -East, z -Down) (right-handed) coordinate system, or ‘NED’ (cf. Additional file 1: Text S2, Fig. S1). We provide examples of axis alignment, outline the importance of transforming between coordinate frames (relative to the Earth’s fixed frame) and recommend a universal configuration calibration procedure to aid correct axis alignment within Additional file 1: Text S2.

Multiple mathematically sophisticated algorithms have been developed to correct distortions from each magnetometer channel’s output [e.g., 23, 149, 150, 151, 152]. We provide an annotated R script—*Gundog Compass* (Additional file 2) that corrects both soft and hard iron distortions from tri-axial magnetometry data



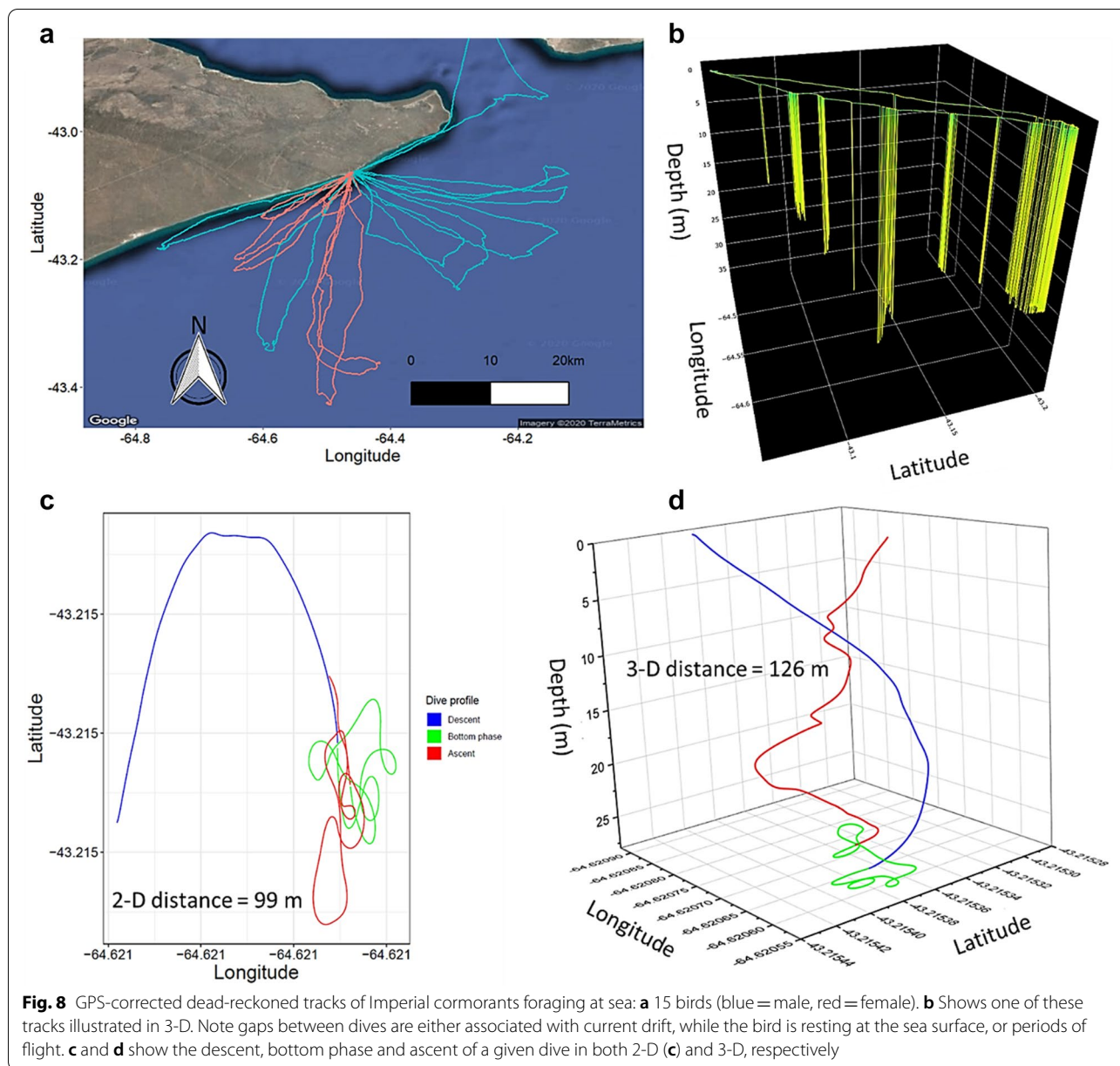


Fig. 8 GPS-corrected dead-reckoned tracks of Imperial cormorants foraging at sea: **a** 15 birds (blue = male, red = female). **b** Shows one of these tracks illustrated in 3-D. Note gaps between dives are either associated with current drift, while the bird is resting at the sea surface, or periods of flight. **c** and **d** show the descent, bottom phase and ascent of a given dive in both 2-D (**c**) and 3-D, respectively

and subsequently computes tilt-compensated heading (0° to 360°). Within this function, there are two main methods of correction to choose from, based on the mathematical protocols outlined by Vitali [153]—least-square error approximation (constructing an ellipsoid rotation matrix) and Winer [154]—scale biases with simple orthogonal rescaling (avoiding matrices altogether). We expand on this user-defined functionality, as well as outlining the causes of soft and hard iron distortions and the initial calibration procedure required to correct such distortions within Additional file 1: Text S3.

Tilt-compensated heading derivation

Device orientation is expressed in terms of a sequence of Euler angle [roll (Φ), pitch (θ), yaw (Ψ)] rotations about the x -, y - and z -axes, respectively, relative to the (inertial) Earth's fixed frame of reference (e.g., Earth-Centre, Earth-Fixed (ECEF) system) [155]. Being a vector field sensor with two degrees of rotational freedom, accelerometers are insensitive to rotations about the gravity vector and thus discerning heading requires the arctangent of the ratio between the x - and y -orthogonal magnetometer measurements [156]. For the correct computation of heading, these two channels need to be aligned parallel to

the earth's surface. This is achieved by correcting any orientation (de-rotation) according to pitch and roll angles (postural offsets) which can be deduced from acceleration. These angles are typically approximated by deriving gravity-based (static) acceleration [see 72, 157] from each channel by employing one of four approaches using: (i) a running mean [e.g., 72, 86], (ii) a Fast-Fourier transformation [e.g., 158], (iii) a high-pass filter [e.g., 159] or (iv) a Kalman-filter [e.g., 160]. Here, we use a computationally simple running mean over 2 s [72] (Eq. 1):

$$G_{x,y,z} = \frac{1}{w} \sum_{j=i-\frac{w}{2}}^{i+\frac{w}{2}} A_{x,y,z} \quad (1)$$

where w is an integer specifying the window size and $G_{x,y,z}$ and $A_{x,y,z}$ represents the smoothed and raw components of acceleration, respectively. In the absence of linear (dynamic) acceleration [see 157, 161], values of $G_{x,y,z}$ reflect the device orientation with respect to the earth's reference frame (though see Table. 1), reading approx. +1 g when orientated directly towards the gravity vector (down), -1 g against the gravity vector (up) and 0 g at perpendicular to it (horizontal). In R, the 'zoo' package [126] provides useful wrappers to apply arithmetic operations in a rolling fashion (R_{1,4}).

```
(R1) install.packages("zoo") ;
      library(zoo)
(R2) Gx = rollapply(Ax, width=w,
                  FUN=mean, align="center",
                  fill="extend")
(R3) Gy = rollapply(Ay, width=w,
                  FUN=mean, align="center",
                  fill="extend")
(R4) Gz = rollapply(Az, width=w,
                  FUN=mean, align="center",
                  fill="extend")
```

Here, w should be replaced with the window width of choice (e.g., for 20 Hz data and a smoothing of 2 s required, replace w with 40). We use a centre-aligned index (compared to the rolling window of observations), with "extend" to indicate repetition of the leftmost or rightmost non-NA value (though fill can equally be set as NA, 0, etc.).

Importantly, for correct trigonometric formulae output within the tilt-compensated compass method, the vectorial sum of static acceleration ($G_{x,y,z}$) and calibrated magnetometry ($M_{x,y,z}$) measurements across all three spatial-dimensions must be normalised (to a unit vector) with a scaled magnitude (radius) of one (Eqs. 2, 3, R_{5,10}). It was previously demonstrated that, for fast moving

animals, high frequency of body posture changes could cause discrepancy between static acceleration data and magnetism data, which could consequently affect heading estimation [162]. Although this effect would not change general shapes of movement paths, we suggest that prior to the normalisation process (and magnetic calibration procedure), it may be of value to initially smooth out (see Eq. 1, R_{1,4}) small deviations within magnetometry data, both to avoid this type of error and to reduce the magnitude of anomalous spikes in magnetic inference. We used a smoothing window of 10 events for the 40 Hz datasets used in this study.

$$\begin{bmatrix} NG_x \\ NG_y \\ NG_z \end{bmatrix} = \frac{1}{\sqrt{G_x \cdot G_x + G_y \cdot G_y + G_z \cdot G_z}} \cdot \begin{bmatrix} G_x \\ G_y \\ G_z \end{bmatrix} \quad (2)$$

$$\begin{bmatrix} NM_x \\ NM_y \\ NM_z \end{bmatrix} = \frac{1}{\sqrt{M_x \cdot M_x + M_y \cdot M_y + M_z \cdot M_z}} \cdot \begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix} \quad (3)$$

```
(R5) NGx = Gx / sqrt(Gx^2 + Gy^2 + Gz^2)
(R6) NGy = Gy / sqrt(Gx^2 + Gy^2 + Gz^2)
(R7) NGz = Gz / sqrt(Gx^2 + Gy^2 + Gz^2)
(R8) NMx = Mx / sqrt(Mx^2 + My^2 + Mz^2)
(R9) NMy = My / sqrt(Mx^2 + My^2 + Mz^2)
(R10) NMz = Mz / sqrt(Mx^2 + My^2 + Mz^2)
```

Depending on deployment position, the device-carried NED coordinate frame (the x -, y -, z -axes) may not correspond with the animal's body-carried NED frame. When this occurs, prior to deriving animal orientation, the normalised gravity and magnetic vectors are required to be corrected so that their measurements are expressed relative to the body frame of the animal [45]. This requires three rotation sequences, using 3 by 3 rotation matrices (Eqs. 4, 6) and involves two intermediate frames. The aerospace sequence used here is as follows:

1. A right-handed rotation (C), about the z -axis axis of the device's frame (D), through angle Ψ (Eq. 4), to get to the first intermediate frame (F_1).
2. A right-handed rotation (C) about the y -axis at F_1 , through angle θ (Eq. 5), to get to the second intermediate frame (F_2).

3. A right-handed rotation (C) about the x -axis at F_2 , through angle Φ (Eq. 6), to get to the animal's body frame (B).

$$C_{F_1/D}^{(\Psi)} = \begin{bmatrix} \cos(\Psi) & \sin(\Psi) & 0 \\ -\sin(\Psi) & \cos(\Psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4)$$

$$C_{F_2/F_1}^{(\theta)} = \begin{bmatrix} \cos(\theta) & 0 & -\sin(\theta) \\ 0 & 1 & 0 \\ \sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \quad (5)$$

$$C_{B/D}^{(\Phi,\theta,\Psi)} = \begin{bmatrix} \cos(\Psi) \cdot \cos(\theta) & \sin(\Psi) \cdot \cos(\theta) & -\sin(\theta) \\ \cos(\Psi) \cdot \sin(\theta) \cdot \sin(\Phi) - \sin(\Psi) \cdot \cos(\Phi) & \sin(\Psi) \cdot \sin(\theta) \cdot \sin(\Phi) + \cos(\Psi) \cdot \cos(\Phi) & \cos(\theta) \cdot \sin(\Phi) \\ \cos(\Psi) \cdot \sin(\theta) \cdot \cos(\Phi) + \sin(\Psi) \cdot \sin(\Phi) & \sin(\Psi) \cdot \sin(\theta) \cdot \cos(\Phi) - \cos(\Psi) \cdot \sin(\Phi) & \cos(\theta) \cdot \cos(\Phi) \end{bmatrix} \quad (8)$$

$$C_{B/F_2}^{(\Phi)} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\Phi) & \sin(\Phi) \\ 0 & -\sin(\Phi) & \cos(\Phi) \end{bmatrix} \quad (6)$$

Note the right-handed rule of rotation; a positive Ψ reflects a clockwise rotation of the anterior–posterior axis (relative to North), a positive θ reflects a nose-upward tilt of this axis and a positive Φ reflects a bank angle tilt to the right about this axis. Reversing the direction of two axes causes a 180° inversion about the remaining axis and interchanging two axes (e.g., x with y) or reversing the direction of one or all three axes reverses the ‘handedness’ of rotation [right-handed—‘counter-clockwise’ vs. left-handed—‘clockwise’ (when viewed from the tip of the z -axis)]. Rotation matrices are orthogonal (unitary), with every row and column being linearly independent and normal to every other row and column. The consequence of this is that the inverse of a rotation matrix is its transpose [163] [which essentially reverses the direction of rotation, and within (Eqs. 4, 6), this is achieved by negating the sign of the sines]. Importantly, because rotation matrices are not symmetric, the order of matrix multiplication is important [45] (otherwise, Euler angles are without meaning for describing orientation).

The product of the conventionally used aerospace rotation sequence outlined above (to get from the tag frame to the animal's body frame) can be expressed as (Eq. 7).

$$C_{B/D}^{(\Phi,\theta,\Psi)} = C_{B/F_2}^{(\Phi)} \cdot C_{F_2/F_1}^{(\theta)} \cdot C_{F_1/D}^{(\Psi)} \quad (7)$$

When matrix multiplied out, this yields (Eq. 8)—often referred to as a Direction Cosine Matrix (DCM). The composition of this DCM varies according the (six possible) orderings of the three rotation matrices (Eqs. 4, 6) and the direction of intended rotation relative to the direction of measured g within the NED system (see Additional file 1: Text S2).

Note the left-handed rule of reading the vectorial notation of ordered rotations, for example $C_{B/D}^{(\Phi,\theta,\Psi)}$ means going from the device frame to the animal's body frame, by first rotating about the z -axis (though angle Ψ), followed by the y -axis (though angle θ) and then lastly the x -axis (though angle Φ). The device offset can be estimated from direct observation or deduced using photographs or from the tag data itself. For example, assuming that ‘normal animal posture’ has no pitch and roll angle offset, then a tri-axial spherical plot of static acceleration [164] would show a densely populated band of datapoints at the cross-sectional origin of 0 g about the x - and y -axes, respectively, when the tag and body NED axes are in alignment.

$NG_{x,y,z}$ and $NM_{x,y,z}$ are pre-multiplied by the DCM to compensate for offset. However, device offset is often parametrised by roll, pitch and/or yaw angles relative to the animal's body frame and thus, the device actually requires de-rotation (switching the ‘handedness’ of rotation) according to these values. For example, a +45° yaw offset requires an inverse rotation about the z -axis by −45°, rather than a further +45° rotation. This simply involves taking the transpose of the DCM (Eq. 9), which is the same as the transpose of each of the individual rotation matrices (Eqs. 10, 11).

$$C_{B/D}^{(\Phi,\theta,\Psi)\top} = \begin{bmatrix} \cos(\Psi) \cdot \cos(\theta) & \cos(\Psi) \cdot \sin(\theta) \cdot \sin(\Phi) - \sin(\Psi) \cdot \cos(\Phi) & \cos(\Psi) \cdot \sin(\theta) \cdot \cos(\Phi) + \sin(\Psi) \cdot \sin(\Phi) \\ \sin(\Psi) \cdot \cos(\theta) & \sin(\Psi) \cdot \sin(\theta) \cdot \sin(\Phi) + \cos(\Psi) \cdot \cos(\Phi) & \sin(\Psi) \cdot \sin(\theta) \cdot \sin(\Phi) - \cos(\Psi) \cdot \sin(\Phi) \\ -\sin(\theta) & \cos(\theta) \cdot \sin(\Phi) & \cos(\theta) \cdot \cos(\Phi) \end{bmatrix} \quad (9)$$

$$\begin{bmatrix} \text{NGb}_x \\ \text{NGb}_y \\ \text{NGb}_z \end{bmatrix}^B = C_{B/F_2}^{(\Phi)T} \cdot C_{F_2/F_1}^{(\theta)T} \cdot C_{F_1/D}^{(\Psi)T} \cdot \begin{bmatrix} \text{NG}_x \\ \text{NG}_y \\ \text{NG}_z \end{bmatrix}^D \quad (10)$$

$$\begin{bmatrix} \text{NMB}_x \\ \text{NMB}_y \\ \text{NMB}_z \end{bmatrix}^B = C_{B/F_2}^{(\Phi)T} \cdot C_{F_2/F_1}^{(\theta)T} \cdot C_{F_1/D}^{(\Psi)T} \cdot \begin{bmatrix} \text{NM}_x \\ \text{NM}_y \\ \text{NM}_z \end{bmatrix}^D \quad (11)$$

where T is the matrix transpose and resultant $\text{NGb}_{x,y,z}$ and $\text{NMB}_{x,y,z}$ vectors are expressed in the animal's body-carried NED frame. The input of these gravity and magnetic vectors are supplied as 3 by 1 column matrices for true matrix multiplication, and when expanding out (Eq. 11), this results in (Eq. 12) [substituting NM with NG expands out (Eq. 10)].

$$\begin{bmatrix} \text{NMB}_x \\ \text{NMB}_y \\ \text{NMB}_z \end{bmatrix}^B = \begin{bmatrix} \text{NM}_x \cdot \cos(\Psi) \cdot \cos(\theta) + \text{NM}_y \cdot (\cos(\Psi) \cdot \sin(\theta) \cdot \sin(\Phi) - \sin(\Psi) \cdot \cos(\Phi)) + \text{NM}_z \cdot (\cos(\Psi) \cdot \sin(\theta) \cdot \cos(\Phi) + \sin(\Psi) \cdot \sin(\Phi)) \\ \text{NM}_x \cdot \sin(\Psi) \cdot \cos(\theta) + \text{NM}_y \cdot (\sin(\Psi) \cdot \sin(\theta) \cdot \sin(\Phi) + \cos(\Psi) \cdot \cos(\Phi)) + \text{NM}_z \cdot (\sin(\Psi) \cdot \sin(\theta) \cdot \sin(\Phi) - \cos(\Psi) \cdot \sin(\Phi)) \\ -\text{NM}_x \cdot \sin(\theta) + \text{NM}_y \cdot \cos(\theta) \cdot \sin(\Phi) + \text{NM}_z \cdot \cos(\theta) \cdot \cos(\Phi) \end{bmatrix}^D \quad (12)$$

In R then, the alignment of device to body axes for both gravity and magnetic vectors can be performed using the following procedure (R_{11:22}).

- (R11) RollSinAngle = sin(Roll * pi/180)
- (R12) RollCosAngle = cos(Roll * pi/180)
- (R13) PitchSinAngle = sin(Pitch * pi/180)
- (R14) PitchCosAngle = cos(Pitch * pi/180)
- (R15) YawSinAngle = sin(Yaw * pi/180)
- (R16) YawCosAngle = cos(Yaw * pi/180)
- (R17) $\text{NGb}_x = \text{NG}_x * \text{YawCosAngle} * \text{PitchCosAngle} + \text{NG}_y * (\text{YawCosAngle} * \text{PitchSinAngle} * \text{RollSinAngle} - \text{YawSinAngle} * \text{RollCosAngle}) + \text{NG}_z * (\text{YawCosAngle} * \text{PitchSinAngle} * \text{RollCosAngle} + \text{YawSinAngle} * \text{RollSinAngle})$
- (R18) $\text{NGb}_y = \text{NG}_x * \text{YawSinAngle} * \text{PitchCosAngle} + \text{NG}_y * (\text{YawSinAngle} * \text{PitchSinAngle} * \text{RollSinAngle} + \text{YawCosAngle} * \text{RollCosAngle}) + \text{NG}_z * (\text{YawSinAngle} * \text{PitchSinAngle} * \text{RollSinAngle} - \text{YawCosAngle} * \text{RollSinAngle})$
- (R19) $\text{NGb}_z = -\text{NG}_x * \text{PitchSinAngle} + \text{NG}_y * \text{PitchCosAngle} * \text{RollSinAngle} + \text{NG}_z * \text{PitchCosAngle} * \text{RollCosAngle}$
- (R20) $\text{NMB}_x = \text{NM}_x * \text{YawCosAngle} * \text{PitchCosAngle} + \text{NM}_y * (\text{YawCosAngle} * \text{PitchSinAngle} * \text{RollSinAngle} - \text{YawSinAngle} * \text{RollCosAngle}) + \text{NM}_z * (\text{YawCosAngle} * \text{PitchSinAngle} * \text{RollCosAngle} + \text{YawSinAngle} * \text{RollSinAngle})$

$$\text{PitchSinAngle} * \text{RollSinAngle} - \text{YawSinAngle} * \text{RollCosAngle) + NM}_z * (\text{YawCosAngle} * \text{PitchSinAngle} * \text{RollCosAngle} + \text{YawSinAngle} * \text{RollSinAngle})$$

$$\text{(R21) NM}_y = \text{NM}_x * \text{YawSinAngle} * \text{PitchCosAngle} + \text{NM}_y * (\text{YawSinAngle} * \text{PitchSinAngle} * \text{RollSinAngle} - \text{YawSinAngle} * \text{RollCosAngle}) + \text{NM}_z * (\text{YawSinAngle} * \text{PitchSinAngle} * \text{RollSinAngle} - \text{YawCosAngle} * \text{RollSinAngle})$$

$$\text{(R22) NM}_z = -\text{NM}_x * \text{PitchSinAngle} + \text{NM}_y * \text{PitchCosAngle} * \text{RollSinAngle} + \text{NM}_z * \text{PitchCosAngle} * \text{RollCosAngle}$$

Here, *Roll*, *Pitch* and *Yaw* inputs denote the angular off-

set of the device, relative to the animal body frame. Note, standard trigonometric functions operate in radians, not degrees. In base R, $\pi = \text{pi}$. Multiplying values by $\text{pi}/180$ converts degrees into radians, whilst multiplying values by $180/\text{pi}$ does the reverse. This rotation correction procedure is implemented within *GundogCompass* when pitch, roll and/or yaw offsets are supplied (Additional file 2).

Following the alignment of device and body axes, pitch and roll of the animal are calculated from the DCM, and because there are multiple variations in the order that rotation sequences can be composed and applied, there are also different valid equations that output different pitch and roll angle estimates, for equivalent static acceleration input. The convention is to use formulae that have no dependence on yaw rotation and restrict either the pitch or the roll angles within the range -90° to $+90^\circ$ (but not both), with the other axis of rotation able to lie between -180° and 180° , thereby eliminating duplicate solutions at multiples of 360° . Multiplying (Eq. 8) by the measured Earth's gravitational field vector ($+1 \text{ g}$ when initially aligned downwards along the z-axis) simplifies down to (Eq. 13). The accelerometer output for this aerospace rotation sequence is thus only dependent on the roll and pitch angles which can be solved (Eqs. 14, 15), allowing roll angles the greater freedom [161]. This is relevant for studies using collar-mounted tags, whereby collar may roll $>90^\circ$ in either direction from default orientation.

$$\text{NGb}_{xyz} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\Psi) \cdot \cos(\theta) & \sin(\Psi) \cdot \cos(\theta) & -\sin(\theta) \\ \cos(\Psi) \cdot \sin(\theta) \cdot \sin(\Phi) - \sin(\Psi) \cdot \cos(\Phi) & \sin(\Psi) \cdot \sin(\theta) \cdot \sin(\Phi) + \cos(\Psi) \cdot \cos(\Phi) & \cos(\theta) \cdot \sin(\Phi) \\ \cos(\Psi) \cdot \sin(\theta) \cdot \cos(\Phi) + \sin(\Psi) \cdot \sin(\Phi) & \sin(\Psi) \cdot \sin(\theta) \cdot \cos(\Phi) - \cos(\Psi) \cdot \sin(\Phi) & \cos(\theta) \cdot \cos(\Phi) \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} -\sin(\theta) \\ \cos(\theta) \cdot \sin(\Phi) \\ \cos(\theta) \cdot \cos(\Phi) \end{bmatrix} \quad (13)$$

$$\tan \theta_{xyz} = \left(\frac{-\text{NGb}_x}{\sqrt{\text{NGb}_y^2 + \text{NGb}_z^2}} \right) \Rightarrow \theta = \text{atan2} \left(-\text{NGb}_x, \sqrt{(\text{NGb}_y \cdot \text{NGb}_y + \text{NGb}_z \cdot \text{NGb}_z)} \right) \cdot \frac{180}{\pi} \quad (14)$$

$$\tan \Phi_{xyz} = \left(\frac{\text{NGb}_y}{\text{NGb}_z} \right) \Rightarrow \Phi = \text{atan2}(\text{NGb}_y, \text{NGb}_z) \cdot \frac{180}{\pi} \quad (15)$$

The equation for roll (Eq. 15), however, has a region of instability at obtuse pitch angles (e.g., for NED systems, the x -axis points directly up or down, with respect to the Earth's frame of reference). Whilst there is no 'gold standard' solution to this problem of singularity (using Euler angles), an attractive circumvention (detailed within [161]) is to modify (Eq. 15) and add a very small percentage (μ) of the NGb_x reading into the denominator, preventing it ever being zero and thus driving roll angles to zero when pitch approaches $-/+90^\circ$ for stability (Eq. 16).

$$\begin{bmatrix} \text{NMBf}_x \\ \text{NMBf}_y \\ \text{NMBf}_z \end{bmatrix} = \begin{bmatrix} \text{NMB}_x \cdot \cos(\theta) + \text{NMB}_y \cdot \sin(\theta) \cdot \sin(\Phi) + \text{NMB}_z \cdot \sin(\theta) \cdot \cos(\Phi) \\ \text{NMB}_y \cdot \cos(\Phi) - \text{NMB}_z \cdot \sin(\Phi) \\ -\text{NMB}_x \cdot \sin(\theta) + \text{NMB}_y \cdot \cos(\theta) \cdot \sin(\Phi) + \text{NMB}_z \cdot \cos(\theta) \cdot \cos(\Phi) \end{bmatrix} \cdot \begin{bmatrix} \text{NMB}_x \\ \text{NMB}_y \\ \text{NMB}_z \end{bmatrix} \quad (17)$$

$$\Phi = \text{atan2}(\text{NGb}_y, \text{sign}(\text{NGb}_z) \cdot \sqrt{(\text{NGb}_z \cdot \text{NGb}_z + \mu \cdot \text{NGb}_x \cdot \text{NGb}_x)}) \cdot \frac{180}{\pi} \quad (16)$$

where $\text{sign}(\text{NGb}_z)$ is allocated the value +1 when NGb_z is non-negative and -1, when NGb_z is negative (recovers directionality of NGb_z , subsequent to the square-root). Taken together then, in R, pitch and roll are computed according to, (R_{24:25}) with outputs within the range of -90° to $+90^\circ$ for pitch and -180° to $+180^\circ$ for roll, and this is the formula we use in the tilt-compensated method outlined below (and within Additional file 2).

```
(R23) mu = 0.01 ; sign = ifelse(NGbz >= 0, 1, -1)
```

```
(R24) Pitch = atan2(-NGbx, sqrt(NGby^2 + NGbz^2)) * 180/pi
```

```
(R25) Roll = atan2(NGby, sign * sqrt(NGbz^2 + mu * NGbx^2)) * 180/pi
```

Here, prior to the derivation of pitch and roll, μ is allocated the value 0.01 and a vector termed 'sign' is created,

containing 1 s and -1 s according to the direction of measured g from NGb_z (R₂₃).

The magnetic vector of the device is then de-rotated to the Earth frame (tilt-corrected) by pre-multiplying by the product of the inverse roll multiplied by inverse pitch rotation matrix (Eq. 17), which when expanded out gives (Eq. 18).

$$\begin{bmatrix} \text{NMBf}_x \\ \text{NMBf}_y \\ \text{NMBf}_z \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \cdot \sin(\Phi) & \sin(\theta) \cdot \cos(\Phi) \\ 0 & \cos(\Phi) & -\sin(\Phi) \\ -\sin(\theta) & \cos(\theta) \cdot \sin(\Phi) & \cos(\theta) \cdot \cos(\Phi) \end{bmatrix} \cdot \begin{bmatrix} \text{NMB}_x \\ \text{NMB}_y \\ \text{NMB}_z \end{bmatrix} \quad (17)$$

$$\begin{bmatrix} \text{NMBf}_x \\ \text{NMBf}_y \\ \text{NMBf}_z \end{bmatrix} = \begin{bmatrix} \text{NMB}_x \cdot \cos(\theta) + \text{NMB}_y \cdot \sin(\theta) \cdot \sin(\Phi) + \text{NMB}_z \cdot \sin(\theta) \cdot \cos(\Phi) \\ \text{NMB}_y \cdot \cos(\Phi) - \text{NMB}_z \cdot \sin(\Phi) \\ -\text{NMB}_x \cdot \sin(\theta) + \text{NMB}_y \cdot \cos(\theta) \cdot \sin(\Phi) + \text{NMB}_z \cdot \cos(\theta) \cdot \cos(\Phi) \end{bmatrix} \quad (18)$$

Here $\text{NMBf}_{x,y,z}$ are the calibrated, normalised magnetometry data (expressed in the animal's body-carried NED frame) after tilt-correction. Finally, yaw (ψ) (heading—now defined by the compass convention, relative to magnetic North) can be computed from the NMBf_x and NMBf_y (Eq. 19) via;

$$\psi = \text{atan2}(-\text{NMBf}_y, \text{NMBf}_x) \cdot \frac{180}{\pi} \quad (19)$$

We outline the R code for this procedure below (R_{26:34}).

```
(R26) RollSinAngle = sin(Roll * pi/180)
```

```
(R27) RollCosAngle = cos(Roll * pi/180)
```

```
(R28) PitchSinAngle = sin(Pitch * pi/180)
```

```
(R29) PitchCosAngle = cos(Pitch * pi/180)
```

```
(R30) NMBfx = NMBx * PitchCosAngle + NMB_y * PitchSinAngle * RollSinAngle +
```

```
NMBz * PitchSinAngle * RollCosAngle
```

```
(R31) NMBfy = NMB_y * RollCosAngle - NMBz * RollSinAngle
```

```
(R32) NMbfz = -NMbx * PitchSinAngle +
      NMby * PitchCosAngle + RollSinAngle +
      NMbz * PitchCosAngle * RollCosAngle
(R33) Yaw = atan2(-NMfby, NMfbx) * 180 /
      pi
(R34) Yaw = ifelse(Yaw < 0, Yaw + 360,
      Yaw)
```

Note, yaw output from (R₃₃) uses the scale -180° to $+180^\circ$. (R₃₄) converts to the scale 0° to 360° (specifically, 0° to 359°). This is also achieved by using a modulus (mod) operator (Eq. 20, R₃₅), which in base R takes the form `%%`.

$$\psi = \text{mod}(360 + \psi, 360) \quad (20)$$

```
(R35) Yaw = (360 + Yaw) %% 360
```

Magnetic declination is defined as the angle on the horizontal plane between magnetic north and true north [165]. Prior to dead-reckoning, magnetic declination should be summed to heading values to convert from magnetic to true North [166]. There are many online sources to calculate the magnetic declination of an area [e.g., 167]. Notably, logical corrections may need to be performed to ensure data does not exceed either circular direction after applying magnetic declination (R₃₆).

```
(R36) h = ifelse(h < 0, h + 360, h) ; h
      = ifelse(h > 360, h - 360, h),
```

where h refers to the vector containing the heading data. Should the user not correct for axis alignment between the device and animal body frame (cf. Eqs. 4–12, R_{11:22}) then a reasonable post-correction for small discrepancies about the yaw axis would be to subtract the difference to h values at this point.

Preparing speed estimates

The vectorial dynamic body acceleration (VeDBA) (Eq. 21) [cf. 67, 168] was our choice of DBA-based speed proxy for terrestrial dead-reckoning purposes. This is given by:

$$v = \sqrt{(D_x^2 + D_y^2 + D_z^2)} \quad (21)$$

where v represents VeDBA, D_x , D_y and D_z are the dynamic acceleration values from each axis, themselves obtained by subtracting each axis' static component of

acceleration (cf. Eq. 1, R_{1:4}) from their raw equivalent (R₃₇).

```
(R37) v = sqrt((Ax - Gx)^2 + (Ay -
      Gy)^2 + (Az - Gz)^2),
```

where A_x , A_y , A_z and G_x , G_y , G_z are the raw and static (smoothed) values of each channel's recorded acceleration.

We recommend implementing a running mean (cf. Eq. 1, R_{1:4}) to raw VeDBA values to ensure that both acceleration and deceleration components of a stride cycle are incorporated together per unit time and to reduce the magnitude of small temporal spikes (likely not attributable to the scale of movement elicited [cf. 97]). Choice of smoothing window size is dependent on the scale of movement being investigated, though as a basic rule, we suggest 1 to 2 s. For similar reasons, it is also worth post-smoothing raw pitch, roll and heading outputs, although heading requires a circular mean (Eqs. 22, 23) [cf. 169]:

$$\bar{\theta}_p = a \tan 2 \left(\frac{1}{n} \sum_{j=i}^n \sin \left(h_j \cdot \frac{\pi}{180} \right), \frac{1}{n} \sum_{j=i}^n \cos \left(h_j \cdot \frac{\pi}{180} \right) \right) \quad (22)$$

$$\bar{h} = \text{mod} \left(360 + \left(\bar{\theta}_p \cdot \frac{180}{\pi} \right), 360 \right) \quad (23)$$

where h_j and \bar{h} are the unsmoothed and smoothed heading values, $\bar{\theta}_p$ the arithmetic mean after converting degrees to cartesian coordinates and mod refers to the modulo operator.

In R, the above formula can be made into a function (R₃₈), to be applied within the 'rollapply' wrapper (replacing 'FUN = mean' with 'FUN = Circ.Avg') (cf. R_{1:4}).

```
(R38) Circ.Avg = function(x) {
      H.East = mean(sin(x * pi / 180))
      H.North = mean(cos(x * pi / 180))
      MH = (atan2(H.East, H.North)) * 180/pi
      MH = (360 + MH) %% 360
      return(MH)
    }
```

Speed (s) can be estimated from VeDBA (v) via (Eq. 24).

$$s = (v \cdot m) + c \quad (24)$$

where m is the multiplicative coefficient and c is a constant [10, 69]. Here, a user can define various bouts of movement from motion sensor data (e.g., via various machine-learning approaches (for review see Farrahi et al. [170]) or the Boolean-based LoCoD method [101]) and/or substrate condition (e.g., via GPS), to be cross-referenced when allocating variants of the speed coefficients. As a simple example, in R, should walking (coded for as 1) and running (coded for as 2) be teased apart from all other (non-moving) data (coded for as 0) within a Marked Events vector (ME), then ME can be used to allocate various m (and if applicable, c) values using simple 'ifelse' statements (R_{39:40}).

```
(R39) m = ifelse(ME == 1, 1.5,
               ifelse(ME == 2, 3.5, 0))
(R40) c = ifelse(ME > 0, 0.1, 0)
```

Here, walking is given an arbitrary coefficient of 1.5 and running, 3.5 with a value of 0.1 for their constants. All other ME values are given a 0 coefficient and 0 constant, which results in no speed at such times, regardless of DBA magnitude.

By-passing DBA as a speed proxy

Dividing the number of steps detected within a given rolling window length (cf. R_{1:4}), by the window length (s) gives an estimated step count per second. This can be converted to speed by multiplying by a distance per step estimate (assuming constant distance travelled between step gaits). We review this further in Additional file 1: Text S4, including a simple peak finder function—*Gundog.Peaks* (Additional file 3) that locates peaks based on local signal maxima, using a given rolling window, with each candidate peak filtered according to whether it surpassed a threshold height (in conjunction with other potential user-defined thresholds). Note, this method can equally be applied to non-terrestrial species, using flipper/tail beats instead, where appropriate.

For diving animals, a proxy for horizontal speed can be obtained based on animal pitch and rate change in depth [47, 116]. Specifically, rate change of depth (Δd) (units in m/s) is divided by the tangent of pitch (θ) (converted from degrees to radians) (Eq. 25):

$$s = \frac{\Delta d}{\tan\left(\theta \cdot \frac{\pi}{180}\right)} \quad (25)$$

Here, resultant speed values need to be made absolute (positive). This calculation is only valid when the direction of movement is the same as the direction of the animal's longitudinal axis (equal pitch assumption) [cf. 47] and

thus should only be calculated at times when the animal is travelling 'ballistically' (at considerable vertical speed).

```
(R41) s = ifelse(abs(p) >= 10, abs(RCD
    / tan(p * pi/180)), s)
```

In the above example (R₄₁), nominal speed values are overwritten with the trigonometric formula output (Eq. 25) at times of 'appreciable' pitch (10°) [cf. 171], where RCD is the rate change of depth and p is the pitch (in radians). An upper limit should be imposed on speed values derived in this way because values can become highly inflated when the pitch angle is particularly acute.

Converting speed to a distance coefficient

Speed (s) estimates are multiplied by the time difference between the values (TD) to give a distance estimate (units in metres) which, in turn, standardises coefficient comparisons across datasets sampled at different rates. These distance values are then divided by the approximate radius of the earth ($R=6,378,137$ m) to give a radial distance coefficient (q) [see 172] (Eq. 26)

$$q = \frac{s \cdot \text{TD}}{R} \quad (26)$$

Assuming that high-resolution depth data are not available, but 'absolute' speed estimates have been obtained, then an alternative to Eq. 25, (in accordance with the equal pitch assumption) is to derive horizontal distance estimates by multiplying the absolute distance by the cosine of the pitch (θ) (converted from degrees to radians), which can equally be performed on the radial distance (Eq. 27):

$$q = q \cdot \cos\left(\theta \cdot \frac{\pi}{180}\right) \quad (27)$$

In R, to determine accurate lengths of time between values, it is best to save date and time variables together as *POSIX* class [173]. Creating timestamp (TS) objects with *POSIXct* class enables greater control and manipulation of time data. This makes computing the rolling time difference (TD; units in seconds) between data points simple (R₄₂):

```
(R42) TD = c(0, difftime(TS, lag(TS),
    units = "secs")[-1])
```

We detail how to create timestamp objects of *POSIXct* class within Additional file 1: Text S5, including formatting with decimal seconds (important for infra-second datasets) and various codes useful for manipulating data to be dead-reckoned based on time.

In R then, following the computation of TD, q is obtained via (R_{43:44}).

$$(R43) s = (v * m) + c$$

$$(R44) q = (s * TD) / 6378137$$

Note, if a negative c intercept is used (e.g., to allow for some body movement without translation), then any negative speed values would need to be equated to zero as an additional step.

As previously mentioned, the ME vector (progressive movement coded by integer values greater than zero (e.g., 1) and stationary behaviour coded by zero) can be used to ensure q (essentially the distance moved) is zero when ME reads zero, ensuring dead-reckoned tracks are not advanced at such times, regardless of the computed speed (R₄₅).

$$(R45) q = \text{ifelse}(ME == 0, 0, q)$$

Derivation of coordinates

Once q and h are obtained, coordinates are advanced using (Eqs. 28, 29);

$$Lat_i = \text{asin}(\sin Lat_0 \cdot \cos q + \cos Lat_0 \cdot \sin q \cdot \cos h) \tag{28}$$

$$Lon_i = Lon_0 + \text{atan2}((\sin h \cdot \sin q \cdot \cos Lat_0), (\cos q - \sin Lat_0 \cdot \sin Lat_i)) \tag{29}$$

where Lat_0 , Lat_i and Lon_0 , Lon_i are the previous and present latitude and longitude coordinates, respectively (in radians), h is the (present) heading (in radians) and q is the (present) distance coefficient.

In R, the above can be performed iteratively within a for-loop (iteration of code repeated per consecutive i th element of data; R₄₉). Initialising the output latitude ($DR.lat$) and longitude ($DR.lon$) variables to the required length (e.g., as governed by the vector length of other input data (heading, speed, etc.) speeds up processing time (R₄₆). Within the trigonometric dead-reckoning formulae, the starting latitude (la) and longitude (lo) coordinates and heading (h) values must be supplied in radians (R₄₇). The la and lo values are saved as the first elements of the $DR.lat$ and $DR.lon$ vectors to be advanced, respectively (R₄₈).

```
(R46) DR.lat = rep(NA, length(h)) ;
      DR.lon = rep(NA, length(h))
(R47) la = la * pi/180 ; lo = lo *
      pi/180 ; h = h * pi/180
(R48) DR.lat[1] = la DR.lon[1] = lo
(R49) for(i in 2:length(DR.lat)) {
DR.lat[i] = asin(sin(DR.lat[i-1]) *
      cos(q[i]) n+ cos(DR.lat[i-1]) *
      sin(q[i]) * cos(h[i]))
```

```
DR.lon[i] = DR.lon[i-1] +
      atan2(sin(h[i]) * sin(q[i]) *
      cos(DR.lat[i-1]), cos(q[i]) - sin(DR.
      lat[i-1]) * sin(DR.lat[i]))
}
```

Reverse dead-reckoning

For this, firstly, the time difference is computed as usual (R₅₀) and the dimensions of each vector required in the dead-reckoning calculation are reversed. We bind all relevant vectors into a data frame (df) (R₅₁), subsequent to reversing data frame dimensions (R₅₂); the last row becomes the first row, second to last row becomes the second, etc. Note, this can equally be achieved by using the $rev()$ function within base R, on each individual vector. These reversed columns are now restored as vectors (R₅₃) and shifted forward by one element (R₅₄). This is required for correct alignment in time so that dead-reckoning works in exactly the opposite manner to ‘forward’ dead-reckoning.

```
(R50) TD = c(0, difftime(TS, lag(TS),
      units = "secs")[-1])
(R51) df = data.frame(TD, h, v, m, c,
      ME)
(R52) df = df[dim(df)[1]:1, ]
(R53) TD = df[, 'TD'] ; h = df[, 'h'] ;
      v = df[, 'v'] ;
m = df[, 'm'] ; c = df[, 'c'] ; ME =
      df[, 'ME']
(R54) TD = c(NA, TD[-length(TD)]) ; h =
      c(NA, h[-length(h)]) ;
v = c(NA, v[-length(v)]) ; m = c(NA,
      m[-length(m)]) ;
c = c(NA, c[-length(c)]) ; ME = c(NA,
      ME[-length(ME)])
```

The next step is to rotate heading 180° and correct for its circular nature (R₅₅).

```
(R55) h = h - 180 ; h = ifelse(h < 0, h
      + 360, h)
```

Lastly, q is determined and $DR.lon$ and $DR.lat$ are advanced based on the dead-reckoning formula (cf. R_{46:49}), except in this instance, the first element of $DR.lon$ and $DR.lat$ needs to be supplied by the ‘known’ last lo and la coordinates.

Integrating current vectors

In R, current vectors can be added according to (R_{56:60}). Current speed (cs) is in m/s (ensure values are absolute) and current heading (ch) uses the scale 0° to 360°. Note the use of ‘ y ’ and ‘ xx ’ vectors, storing the previous $DR.lat$ and $DR.lon$ coordinates prior to implementing the

next ‘current drift’ vector per iteration. The current speed is also standardised according to the time period length

estimates from motion data—though they are essentially the same).

$$d = 2 \cdot R \cdot \sin^{-1} \left(\sqrt{\sin^2 \left(\frac{\text{Lat}_i - \text{Lat}_0}{2} \right) + \cos(\text{Lat}_0) \cdot \cos(\text{Lat}_i) \cdot \sin^2 \left(\frac{\text{Lon}_i - \text{Lon}_0}{2} \right)} \right) \tag{30}$$

and Earth’s radius (analogous to the derivation of q). When reverse dead-reckoning, it is important to ensure that cs and ch are included in the steps outlined above (R_{50:55}).

```
(R56) DR.lat = rep(NA, length(h)) ;
      DR.lon = rep(NA, length(h))
(R57) xx <- rep(NA, length(cs)) ; yy <-
      rep(NA, length(cs))
(R58) la = la * pi/180 ; lo = lo *
      pi/180 ;
h = h * pi/180 ; ch = ch * pi/180
(R59) DR.lat[1] = la DR.lon[1] = lo
(R60) for(i in 2:length(DR.lat)) {
  DR.lat[i] = asin(sin(DR.lat[i-1]) *
    cos(q[i]) + cos(DR.lat[i-1]) *
    sin(q[i]) * cos(h[i]))
  yy[i] = DR.lat[i]
  DR.lon[i] = DR.lon[i-1] +
    atan2(sin(h[i]) * sin(q[i]) *
    cos(DR.lat[i-1]), cos(q[i]) - sin(DR.
    lat[i-1]) * sin(DR.lat[i]))
  xx[i] = DR.lon[i]
  DR.lat[i] = asin(sin(yy[i]) *
    cos((cs[i] * TD[i]) / 6378137) +
    cos(yy[i]) * sin((cs[i] * TD[i]) /
    6378137) * cos(ch[i]))
  DR.lon[i] = xx[i] + atan2(sin(ch[i]) *
    sin((cs[i] * TD[i]) /
    6378137) * cos(yy[i]), cos((cs[i] *
    TD[i]) / 6378137) - sin(yy[i]) *
    sin(DR.lat[i]))
}
```

VPC procedure

Specifically, this method entails calculating the difference of Haversine distance (net error) and bearing (from true North) between consecutive VPs and the corresponding time-matched dead-reckoned track positions. The trigonometric Haversine formulae (Eqs. 30, 31) are used to calculate the great-circle distance (d) and great circular bearing (b) between consecutive VPs and consecutive (time-matched) dead-reckoned positions (note we use the term ‘bearing’ to differentiate between heading

where R is the Earth’s radius and d , the output in metres.

$$b = \text{atan2} \left(\frac{\sin(\Delta\text{Lon}) \cdot \cos(\text{Lat}_i), \cos(\text{Lat}_0) \cdot \sin(\Delta\text{Lat}) \cdot \cos(\text{Lat}_i) \cdot \cos(\Delta\text{Lon}) \right) \cdot \frac{180}{\pi} \tag{31}$$

where ΔLon represents $\text{Lon}_i - \text{Lon}_0$, ΔLat represents $\text{Lat}_i - \text{Lat}_0$ and b output is in the scale -180° to $+180^\circ$. To convert b to the conventional 0° to 360° scale, 360 should be added to values < 0 .

For each VP, the distance is divided by the dead-reckoned distance providing a distance correction factor (ratio; Eq. 32). The heading correction factor is computed by subtracting the dead-reckoned bearing from the VP bearing (Eq. 33). To ensure that difference does not exceed 180° in either circular direction, 360 should be added to values < -180 and 360 subtracted from values > 180 . A simple example of why this is relevant can be illustrated by subtracting a dead-reckoned bearing value of 359° from a VP bearing value of 1° —post-correction, the difference is $+2^\circ$.

$$\text{Distance}_{\text{corr.factor}} = \frac{\text{Distance}_{\text{VP}}}{\text{Distance}_{\text{DR}}} \tag{32}$$

$$\text{Heading}_{\text{corr.factor}} = \text{Bearing}_{\text{VP}} - \text{Bearing}_{\text{DR}} \tag{33}$$

All intermediate q values are multiplied by the distance correction factor and the heading correction factor is added to all intermediate h values (ensuring that h values are in degrees). To ensure circular range is maintained between 0° and 360° , 360 should be subtracted from values > 360 and added to values < 0 .

Specifically, we follow the protocol illustrated within Fig. 9 for intermediate values. Note the formulae to calculate both distance (d ; Eq. 30) and bearing (b ; Eq. 31) between two points, are also used to recalculate both the heading (h) and radial distance (q) between current-integrated dead-reckoned fixes (pre-VPC; cf. R₆₀). Note that the Haversine distance is required to be converted back to radial distance by dividing by R ($R = 6,378,137$).

In R, the formulae to calculate the great-circle distance and great circular bearing are saved within the *disty* (R₆₁) and *beary* (R₆₂) functions, respectively, where lon1 , lat1 , long2 and lat2 represent longitude and latitude positions (decimal format) at t_i and t_{i+1} , (t representing time).

GPS fix	DR fix	Intermediate Correction order	ΔLon ΔLat
1*	1*	—	
	2	1	
	3	1	
	4	1	
2	5	1	
	6	2	
	7	2	
	8	2	
	9	2	
	10	2	
3	11	2	
	12	3	
	13	3	
4	14	3	
	15	4	
	16	4	
	17	4	
	18	4	
	19	4	
5	20	4	
	21	5	

Fig. 9 Schematic diagram illustrating the order of fixes used when calculating the $Distance_{corr.factor}$ and $Heading_{corr.factor}$ (difference of both GPS and dead-reckoned (DR) positions between arrow heads). Note the discrepancy with the order at which these correction factors are applied to intermediate DR positions (as denoted by colour shading). Known starting position denoted with asterisk

```
(R61) disty = function(long1, lat1,
    long2, lat2) {
    long1 = long1 * pi/180 ; long2 =
        long2 * pi/180 ; lat1 = lat1 *
    pi/180 ; lat2 = lat2 * pi/180
    a = sin((lat2 - lat1) / 2) * sin((lat2
        - lat1) / 2) + cos(lat1) *
    cos(lat2) * sin((long2 - long1) / 2) *
        sin((long2 - long1) / 2)
    c = 2 * atan2(sqrt(a), sqrt(1 - a))
```

```
d = 6378137 * c
    return(d)
}
(R62) beary = function(long1, lat1,
    long2, lat2) {
    long1 = long1 * pi/180 ; long2 = long2
        * pi/180 ; lat1 = lat1 * pi/1
    80 ; lat2 = lat2 * pi/180
    a = sin(long2 - long1)*cos(lat2)
    b = cos(lat1) * sin(lat2) - sin(lat1)
        * cos(lat2) * cos(long2 - long1)
    c = ((atan2(a, b) / pi)*180)
    return(c)
}
```

Below, we outline an example of VPC in R and assume VP coordinates (decimal format) are aligned in the same length vectors/columns as motion sensor-derived data, e.g., heading, DBA/speed, etc., with the corresponding indexed (element-/row-wise) time. Typically, motion sensor data are recorded at much higher frequency so that there are many dead-reckoned fixes between sequential VPs. As such, in the example below, we assume NAs are expressed in the VP longitude and latitude fields at times of missing locational data. This approach of synchronising VP—with motion sensor data also applies when integrating current data; assuming *ch* and *cs* are element/row-wise matched to the relevant VP grid node.

Firstly, an indexing row number (*Row.number*) vector, the length of the data used in the dead-reckoning operation (e.g., *h*) is created (R₆₃), which is relevant for merging full-sized and under-sampled data frames together (seen later). Together, the row number, (uncorrected) dead-reckoned longitude and latitude coordinates, VP longitude and latitude coordinates, heading and the radial distance vectors are inputted column-wise into a 'main' data frame, termed 'df' (R₆₄; user-assigned column names of each vector are within quotation marks). This data frame is then filtered removing rows with missing VP data and stored as *df.sub* (R₆₅). This under-sampled data frame thus, row-wise, contains the time-matched dead-reckoned and ground-truthed positions. The VPC process is analogous for reverse dead-reckoned tracks—although *VP.lon* and *VP.lat* must also be reversed [*Row.number* remains in ascending order (not reversed)]. The first element of *VP.lon* and *VP.lat* must be the *lo* and *la*, respectively (or for reverse dead-reckoning, the last element prior to reversing these vectors).

```
(R63) Row.number = rep(1:length(h))
(R64) df = data.frame(Row.number, 'DR.
  longitude' = DR.lon,
  'DR.latitude' = DR.lat, 'VP.longitude'
  = VP.lon,
  'VP.latitude' = VP.lat, h, q)
(R65) df.sub = df[!with(df, is.na(VP.
  longitude) | is.na(VP.latitude))
  ,]
```

Both sets of dead-reckoned and VP coordinates are shifted backwards one row within new columns termed; *DR.loni*, *DR.lati*, *VP.loni*, *VP.lati* (R₆₆). Row-wise, these columns represent the consecutive fix at t_{i+1} with their originals being t_i . This provides the correct format for the inputs required within the *disty* (cf. R₆₁) and *beary* (cf. R₆₂) functions. The distances between consecutive dead-reckoned estimates are stored within the column termed *DR.distance* (R₆₇) and the corresponding

distances between VPs are stored within the column termed *VP.distance* (R₆₈). The *VP.distance* is divided by the *DR.distance* to provide the distance correction factor, termed *Dist.corr.factor* (R₆₉). Importantly here, an *ifelse* statement is incorporated so that *Dist.corr.factor* defaults to zero at times when both *VP.distance* and *DR.distance* are zero (otherwise dividing zero by zero in R produces *NaN*'s).

```
(R66) df.sub$DR.loni = c(df.sub[-1,
  'DR.longitude'], NA)

df.sub$DR.lati = c(df.sub[-1, 'DR.
  latitude'], NA)

df.sub$VP.loni = c(df.sub[-1, 'VP.
  longitude'], NA)

df.sub$VP.lati = c(df.sub[-1, 'VP.
  latitude'], NA)
(R67) df.sub$DR.distance= disty(df.
  sub$DR.longitude,
df.sub$DR.latitude, df.sub$DR.loni,
df.sub$DR.lati)
(R68) df.sub$VP.distance= disty(df.
  sub$VP.longitude,
df.sub$VP.latitude, df.sub$VP.loni,
df.sub$VP.lati)
(R69) df.sub$Dist.corr.factor =
  ifelse(df.sub$VP.distance == 0 &
df.sub$DR.distance == 0, 0, df.sub$VP.
  distance / df.sub$DR.distance)
```

Analogous to the distance correction, the bearings between consecutive dead-reckoned estimates are stored within the column termed *DR.head* (R₇₀) and the corresponding bearings between VPs are stored within the column termed *VP.head* (R₇₁). Logical corrections are performed to convert both to the 0° to 360° scale (R₇₂), *DR.head* is subtracted from *VP.head* providing the heading correction factor, termed *Head.corr.factor* (R₇₃) and further logical corrections are performed to ensure a minimum and maximum difference range between -180° to +180° (R₇₄).

```
(R70) df.sub$DR.head = beary(df.sub$DR.
  longitude,
df.sub$DR.latitude, df.sub$DR.loni,
df.sub$DR.lati)
(R71) df.sub$VP.head = beary(df.sub$VP.
  longitude,
df.sub$VP.latitude, df.sub$VP.loni,
df.sub$VP.lati)
```

```
(R72) df.sub$DR.head = ifelse(df.sub$DR.head < 0,
df.sub$DR.head + 360, df.sub$DR.head)
df.sub$VP.head = ifelse(df.sub$VP.head < 0,
df.sub$VP.head + 360, df.sub$VP.head)
(R73) df.sub$Head.corr.factor =
df.sub$VP.head - df.sub$DR.head
(R74) df.sub$Head.corr.factor =
ifelse(df.sub$Head.corr.factor <
-180,
(df.sub$Head.corr.factor + 360),
df.sub$Head.corr.factor)
df.sub$Head.corr.factor = ifelse(df.sub$Head.corr.factor > 180,
df.sub$Head.corr.factor - 360),
df.sub$Head.corr.factor)
```

Only the relevant columns; *Row.number*, *Dist.corr.factor* and *Head.corr.factor* are preserved (R₇₅) and merged back into the main data frame (*df*) based on the matching row numbers (R₇₆). Both *Dist.corr.factor* and *Head.corr.factor* express NA's between VPs. These are replaced with the most recent non-NA (observations carried forwards; R₇₇). *Dist.corr.factor* and *Head.corr.factor* values are shifted forward by one row (R₇₈) for correct alignment purposes with respect to *h* and *q* values to be adjusted (cf. Fig. 9; R_{79,80}). A logical correction is performed to ensure that a 0° to 360° circular scale is maintained after the heading correction (R₈₁). Note, the *na.locf()* function is required from the 'zoo' package, to replace NA values with the last non-NA value.

```
(R75) df.sub = df.sub[, c('Row.number',
'Dist.corr.factor', 'Head.corr.factor')]
(R76) df = merge(df, df.sub, by = "Row.number", all = TRUE)
(R77) df$Dist.corr.factor =
na.locf(df$Dist.corr.factor)
df$Head.corr.factor = na.locf(df$Head.corr.factor)
(R78) df$Dist.corr.factor = c(NA,
df$Dist.corr.factor[-nrow(df)])
df$Head.corr.factor = c(NA, df$Head.corr.factor[-nrow(df)])
(R79) q = (df$q * df$Dist.corr.factor)
(R80) h = (df$h + df$Head.corr.factor)
(R81) h = ifelse(h > 360, h - 360, h) ;
h = ifelse(h < 0, h + 360, h)
```

These updated coefficients are substituted into the dead-reckoning formula (cf. R_{46,49}) and this process is repeated iteratively (using the updated dead-reckoned

coordinates, heading and radial distance each time) until dead-reckoning fixes accord 'exactly' (*Gundog.Tracks* uses a threshold of 0.01 m) with ground-truthed locations. An important pitfall of the correction process to consider is that dividing 'any' value (e.g., >0) by 0 results in infinite (Inf) values in R. This can arise during the correction process when there is a given distance between consecutive VPs, but no displacement between the according dead-reckoned positions. This can be a consequence of ground-truthing too frequently (typically relevant to high-res GPS studies), where positional noise is more apparent during rest periods [cf. 97] and/or wrongly assigned speed estimates/ME values. *Gundog.Tracks* automatically resamples VPC rate where necessary to avoid Inf values, essentially by changing the VPC rate to avoid using successive VPs at times of no dead-reckoned track advancement. Lastly, *Gundog.Tracks* outputs messages to the user's console, detailing up to six stages of dead-reckoning progression, which includes reporting the maximum distance (units in metres) between dead-reckoned- and ground-truthed positions (used within the VPC procedure) at each iteration of correction and whether automatic VPC resampling due to Inf values occurred.

Conclusion

We have provided a comprehensive, fully integrated application of the dead-reckoning procedure within the framework of the programming language, R, from pre-processing raw tri-axial accelerometry and magnetometry data to VPC dead-reckoning. We have highlighted important considerations to increase the accuracy of the analytical procedure and to avoid misinterpretation of error. We have also supplied extensive Additional files 1, 2, 3, 4 and 5 and supporting functions to aid the process of deriving fine-scale movement paths, including the protocols to correct magnetometry data and derive (tilt-compensated) heading. Importantly, we have demonstrated the value of *Gundog.Tracks*; a multi-functional and user-friendly tool to derive animal movement paths across all media of travel, with detailed input flexibility and output summaries. We suggest the next phase in advancing the utility of animal dead-reckoning includes looking for 'track signatures' that may signify a particular behaviour or reference a particular 'ground-truthed' location. Lastly to advance the utility of *Gundog.Tracks*, we aim to optimise future iterations of the online code to speed up computation time on larger datasets (e.g., sub-second data collected over many months).

Supplementary Information

The online version contains supplementary material available at <https://doi.org/10.1186/s40317-021-00245-z>.

Additional file 1. Methods expanded. **Text S1.** Device set up and capture protocol. **Text S2.** The importance of having the correct coordinate system and axis alignment. **Text S3.** Magnetometer calibration, rotation correction and deriving yaw (heading)—*Gundog.Compass()* explained. **Text S4.** Step counts as a distance estimate—*Gundog.Peaks()* explained. **Text S5.** Time Data in R (POSIXct). **Text S6.** VPC dead-reckoning—*Gundog.Tracks()*—explained.

Additional file 2. *Gundog.Compass()* (.R file).

Additional file 3. *Gundog.Peaks()* (.R file).

Additional file 4. *Gundog.Tracks()* (.R file).

Additional file 5. Step by step guide of using *Gundog.Tracks* (.R file) (to use in conjunction with below).

Additional file 6. Raw sensor and GPS data frame (.txt) of a penguin walking out to sea from its nest.

Acknowledgements

We thank South African National and the Department of Wildlife and National Parks, Botswana, for allowing our research in the Kgalagadi Transfrontier Park. We are grateful to support and kind assistance of the staff and Rangers at the Kgalagadi National Park who were involved with this work, especially Steven Smith, Christa von Elling, Wayne Oppel and Corera Links. Pertaining to the field work carried out in Argentina, we express our gratitude to Andrea Benvenuti, Fabian Gabelli, Monserrat Del Caño, La Chola, Miguel, Estancia El Pedral and Estancia San Lorenzo for assistance in various aspects of the research. We also thank the Instituto de Biología de Organismos Marinos (IBIOMAR-CONICET) for logistical support. HME is funded by an Irish Research Council Government of Ireland postgraduate scholarship.

Authors' contributions

The authors declare no conflict of interest and are in agreement to submit to *Animal Biotelemetry*. RMG conceived the study and RMG and RW wrote the initial draft. PH constructed tag housings for all model species used. Data collection for the lions was led by SF, DG, PV, LVS and AB and assisted by CJT, MFB, DMS, SB, MVR, PH and RMG. Data collection for the penguins was led by FQ and data collection for the cormorants was led by AGL, with assistance from KY, TY, and RMG. LB, MDH, RPW and RMG conceptualised the key considerations underlying the R code procedures and associated case-studies, and RMG wrote the *Gundog* scripts and conducted the analysis of the case-studies. MHT supplied data for the Instantaneous tidal currents of the San Lorenzo region. All authors contributed to manuscript revision and LB, HJW, HME, AGL and LVS contributed to the testing and revision of the R syntax. All authors read and approved the final manuscript.

Funding

This research contributes to the CAASE project funded by King Abdullah University of Science and Technology (KAUST) under the KAUST Sensor Initiative. Fieldwork in the Kgalagadi Transfrontier Park was supported in part by a Department for Economy Global Challenges Research Fund grant to MS. Fieldwork within the Chubut Province was supported in part by the National Agency for Scientific and Technological Promotion of Argentina (PICT 2017-1996 and PICT 2018-1480), and the Grants-in-Aid for Scientific Research from the Japan Society for the Promotion of Science (16K18617).

Availability of data and materials

We provide a step-by-step example R script and an example data file of a Magellanic penguin walking out to sea (with time, raw acceleration and magnetometry data, marked events and aligned GPS positions) to demonstrate some of the key concepts outlined within "VPC dead-reckoning procedure in R" section when dead-reckoning using *Gundog.Tracks* (including the initial calibration of magnetometry data with *Gundog.Compass*). All scripts, and the example penguin data file have been uploaded to GitHub [95] and will be made available if the manuscript is accepted for publication. Online scripts will be continually updated and any queries, suggestions and/or reported bugs should be emailed to the corresponding author.

Declarations

Ethics approval and consent to participate

We thank the Conservation Agency from the Chubut Province, Argentina, for the permits to work at Punta León and Península Valdés protected areas (Disp No. 047/19-SsCyAP). All penguin and cormorant handling procedures were reviewed and approved by the Dirección de Fauna y Flora Silvestre y el Ministerio de Turismo y Áreas Protegidas de la Provincia de Chubut (permits to work at San Lorenzo and Punta León, No. 060/19-DFyFS-MP and No. 047-SsCy/19). Ethical approval was also given by Animal Welfare Ethical Review Body (AWERB), approval number: SU-Ethics-Student-260919/1894, reference: IP-1819-30. Conditions and approvals for lion fieldwork were granted by the Animals Scientific Procedures Act (ASPA) at Queens University of Belfast (QUB-BS-AREC-18-006) and Pretoria University (NAS061-19), permit authorisation was given by South African National Parks (Permit Number SCAM 1550).

Consent for publication

Not applicable.

Competing interests

The authors declare no conflict of interest.

Glossary

Acceleration

The first derivative (rate of change) of an object's velocity with respect to time. Units are expressed as metres per second squared (m/s^2) or in G-forces (g). A single G-force on Earth (though this does vary slightly with elevation) is 9.81 m/s^2 . Tri-axial accelerometers measure acceleration in three orthogonal planes (surge—'anterior–posterior', sway—'medio-lateral' and heave—'dorsal–ventral'). Under non-moving conditions, relative to gravity, the device tilt (pitch and roll) can be calculated directly from raw accelerometry values since they are composed entirely of the static force (gravity). Under linear acceleration, 'moving' forces applied to the device (e.g., due to the animal moving) are superimposed to static readings and as such measured animal acceleration is typically comprised of both a static and dynamic component.

Barometric pressure

Pressure with the Earth's atmosphere, that is a measure of force per unit area, often expressed as standard atmosphere (symbol: atm), defined as 101,325 Pa (1013.25 mbar; $1 \text{ Pa} = 1 \text{ N/m}^2$). The Earth's mean sea-level atmospheric pressure is approx. 1 atm. Barometric pressure decreases with elevation and increases with depth.

Centripetal acceleration

Inertial force caused by circular motion because an object is always

	accelerating when either its direction or magnitude (speed) changes, and in circular motion, the direction changes instantaneously. This can cause the animal to ‘pull g ’ such as at times of banking and cornering very fast.	(ECEF) system	the Earth (this is often simplified to ‘Earth frame of reference’ or ‘Earth’s fixed frame’ in text). Its origin is fixed at the Earth’s centre (the x -axis points towards the intersection of the Earth’s Greenwich Meridian and equatorial plane, the y -axis pointing 90 degrees East of the x -axis and the z pointing north, along the Earth’s rotation axis). Note, this is different to the Earth-Centred Inertial (ECI) system, which is non-rotating (and the x -axis instead always points towards the vernal equinox).
Coordinate frame	In 3-D space, this is a set of three vectors (x -, y -, z -axes) of unit length, perpendicular (orthogonal) to each other.		
Current flow vectors	(or ‘external’ current flow vectors). The heading and speed of tidal-/air-currents.		
Current integration	Adding current flow vectors to dead-reckoned travel vectors.		
De-rotation	Within the tilt-compensated compass framework, this is the conversion of the magnetic vector values through multiplying by the transpose (inverse) of the pitch and then roll rotation matrices.	Equal pitch assumption	The animal moves in the same direction and angle as its anterior–posterior axis (relative to North and the gravity vector, respectively).
Distance correction factor	The 2-D Haversine distance ratio between successive Verified positions (VPs) (used in the VP-correction procedure) and corresponding dead-reckoned positions. This is multiplied to all intermediate (between VPs) radial distance (q) values.	Georeference	Within the dead-reckoning framework this is another term used for carrying out VPC-dead-reckoning, or drift-correction or GPS-corrected dead-reckoning.
Drift	The accumulation of spatial errors relative to a Verified Position, arising from integrating incorrect dimensions of travel.	Gimbal lock	This is the loss of a degree of freedom in 3-D, when two axes become parallel to each other (locked in the same attitude, reflecting the same rotation). For example if the anterior–posterior axis (‘surge’ or ‘forward-back’— x -axis for NED coordinate frames) points in the plane of the gravity vector (pitched 90 degrees up or down), then the dorsal–ventral (‘heave’ or ‘up-down’— z -axis for NED coordinate frames) and the medio-lateral axis (‘sway’ or ‘side-to-side’— y -axis for NED coordinate frames) become parallel to each other, and changes about the yaw can no longer be compensated for [changes in the roll (or ‘bank’) is equivalent to changing the heading].
Dynamic body acceleration (DBA)	The dynamic component of acceleration, which is typically induced by the limb and/or spine kinematics of the animal (and thus the attached accelerometer). Generally, more mechanical work (via muscular contraction), corresponds to higher metabolic rate and greater magnitudes in accelerometry readings (dependent on tag deployment site). Typically, dynamic values from each multi-axial channel are integrated into an overall metric, such as ‘Overall Dynamic Body Acceleration’ [ODBA = $ DBAx + DBAy + DBAz $] or ‘Vectorial Dynamic Body Acceleration’ [VeDBA = $(DBAx^2 + DBAy^2 + DBAz^2)^{0.5}$]. Such derivatives have been demonstrated as useful proxies for movement-based power.	GPS-derived speed	The Haversine distance calculated between successive GPS coordinates, divided by the time taken between locations.
Earth-Centre, Earth-Fixed	This defines a non-inertial reference coordinate frame that rotates with	Ground-truthing	Empirical evidence (often information obtained by direct observation), as opposed to inference for validating something under investigation. Within the dead-reckoning framework, VPs such as GPS locations are used to

	periodically ground-truth an animal's position.	system	frame, the origin affixed as the devices centre of gravity and its axes oriented along the geodetic directions defined by the Earth surface (the x - and y -axis pointing true north and East, respectively, parallel to the geoid surface and the z -axis pointing downwards towards the Earth's surface).
Haversine formula	Computes the great-circular distance (units in metres) between two locations (using their longitude and latitude coordinates) on a sphere, applying trigonometry to map a triangle to the surface of a unit sphere. This formula is only an approximation because the Earth is not a perfect sphere, with numerical errors also arising at the antipodal regions.		
Heading correction factor	The difference of heading (or 'bearing') from true North between consecutive VPs (used in the VPC procedure) and temporally aligned dead-reckoned positions. This is summed to all intermediate (between VPs) heading (h) values.	Pitch	The angle of device's anterior–posterior inclination or declination, relative to the horizontal plane of the Earth's surface. Pitch is often expressed as an Euler angle, which describe the attitude and rotations of a device via a given Euler angle sequence (yaw, pitch and roll) of rotations (using rotation matrices). Pitch can be derived from the static component of acceleration. Assuming an NED system, pitch defines the degree of rotation about the y -axis.
Inf	Results from numerical calculations which are mathematically infinite (e.g., in R, dividing any value by zero results in Inf).		
Linear drift correction method	At each path segment, the dead-reckoned path is shifted to the position of the first VP encounter using a shift vector. A correction vector then adds the difference between the VP and estimated dead-reckoned end points linearly over this path segment period.	Radial distance (q)	Within the dead-reckoning framework, this refers to a progression distance accounting for the approximate curvature of the Earth (longlat projection approximating the geoid to a sphere; radius (R) = 6,378,137 m).
Multiplicative (m)-coefficient	Within the dead-reckoning framework, this refers to the gradient of a linear regression, e.g., [speed = (Ve DBA \cdot m) + c], where m is the multiplicative factor of VeDBA and c is the subsequently summed constant value (reflecting the y -intercept).	Right-handed coordinates/rotations	The direction in which the fingers curl when pointing the right thumb along the positive direction (+1 g) of the z -axis (e.g., down for NED coordinates), reflect the direction of rotation to be applied about each axis (for a given Euler angle sequence), with the index finger representing the x -axis and the middle finger representing the y -axis, respectively, when splayed out at right angles to the thumb.
NaN	Non-numeric (un-defined) values (e.g., in R, diving zero by zero results in NaN).		
Net error	Here, net error reflects the 2-D Haversine distance (units in metres) between VPs and temporally aligned dead-reckoned positions.	Roll	The angle of rotation about the device's anterior–posterior axis. Roll is often expressed as an Euler angle, which describe the attitude and rotations of a device via a given Euler angle sequence (yaw, pitch and roll) of rotations (using rotation matrices). Roll is thus derived after rotating by yaw and pitch and can be derived from the static component of acceleration. Assuming an NED system, roll defines the degree of rotation about the x -axis (also termed 'bank
Non-movement behaviours	Behaviour performed while stationary, whereby the animal may be moving, e.g., feeding on the spot, but there is no locomotion (not moving to a different position in 3-D space).		
North–East–Down (NED)	Often used in flight mechanics, this defines a non-inertial 3-D coordinate		

Static body acceleration (SBA)	<p>angle’). The static component of acceleration, due to gravity. Apart from being used to calculate the angle of device tilt, increased inertial (centripetal) acceleration, e.g., when the animal ‘pulls g’, can be captured more fully with static measures (rather than DBA estimates), and analogous to VeDBA, the computation of the Vectorial Static Body Acceleration [VeSBA = $(SBAx^2 + SBAy^2 + SBAz^2)^{0.5}$] has been considered as a proxy of power.</p>	integration	<p>together. Assuming Cartesian coordinates, vector addition is performed by adding the corresponding components of the vectors together. E.g., $[A + B = (a_1 + b_1, a_2 + b_2, \dots, a_n + b_n)]$.</p>
Tilt-compensated method	<p>The compass heading compass (estimated using the arctangent ratio between two orthogonal components of the magnetic vector) is only accurate if the magnetometer outputs [typically x, y channels—assuming the NED coordinate system is used (Additional file 1: Text S2)] are taken when the compass is level. Assuming the accelerometer-magnetometer approach, static acceleration measures are used to calculate the angles between the tag’s gravity (and thus magnetic) vector and the Earth’s frame of reference (e.g., Earth-Centered, Earth-Fixed (ECEF) coordinate system). These angles are typically expressed as pitch and roll Euler angles which are used to compensate for variations in the magnetometer output due to device tilt. The tilt-compensated compass method covers the procedures of adjusting the coordinate frame of the device to correspond with a level inclination and subsequently compute the compass heading from the adjusted magnetometry values.</p>	Vertical speed	<p>Distance travelled vertically up (at altitude) or down (at depth) divided by the time period between values.</p>
Tortuosity	<p>The straight-line distance between the start and end positions of a given path segment, divided by the sum of the consecutive intermediate individual distance steps that constituted the total path segment’s length. Values closer to 0 (or conversely values closer to 1 if subtracting the resultant ‘tortuosity’ value from 1) reflect more twists and turns in the movement path.</p>	World Geodetic (WGS-84)	<p>The typical model of the System 1984 Earth’s shape (standard for maps and satellite navigation), defining a coordinate system that accounts for the oblate spheroid.</p>
Vector	<p>Adding vectors (of travel)</p>	Yaw	<p>The orientation of the device, generally, with respect to true North (assuming any required magnetic declination offset has been applied). Yaw, also termed ‘heading’ or ‘bearing’, is often expressed as an Euler angle, which describe the attitude and rotations of a device via a given Euler angle sequence (yaw, pitch and roll) of rotations (using rotation matrices). Yaw can be derived from the static component of acceleration. Assuming an NED system, yaw defines the degree of rotation about the z-axis. Yaw requires the tilt-compensated compass method to compute.</p>

Author details

¹Swansea Lab for Animal Movement, Department of Biosciences, College of Science, Swansea University, Singleton Park, Swansea SA2 8PP, Wales, UK. ²School of Biological Sciences, Queen’s University Belfast, Belfast, 19 Chlorine Gardens, Belfast BT9 5DL, Northern Ireland, UK. ³Department of Agriculture, Land Reform and Rural Development, Government of South Africa, Pretoria 001, South Africa. ⁴Department of Migration, Max Planck Institute of Animal Behavior, 78315 Radolfzell, Germany. ⁵Department of Veterinary Tropical Diseases, Faculty of Veterinary Science, University of Pretoria, Onderstepoort 0110, South Africa. ⁶School of Biology and Environmental Science, University College Dublin, Belfield, Dublin, Ireland. ⁷Instituto de Biología de Organismos Marinos (IBIOMAR), CONICET, Boulevard Brown 2915, U9120ACD Puerto Madryn, Chubut, Argentina. ⁸Departamento de Ecología, Genética y Evolución & Instituto de Ecología, Genética y Evolución de Buenos Aires (IEGEB), CONICET, Pabellón II Ciudad Universitaria, C1428EGA Buenos Aires, Argentina. ⁹Centre for Biomathematics, College of Science, Swansea University, Swansea SA2 8PP, UK. ¹⁰Graduate School of Environmental Studies, Nagoya University, Furo-cho, Chikusa-ku, Nagoya, Japan. ¹¹Organization for the Strategic Coordination of Research and Intellectual Properties, Meiji University, Nakano, Tokyo, Japan. ¹²Savanna and Grassland Research Unit, South African National Parks, Scientific Services Skukuza, Kruger National Park, Skukuza 1350, South Africa. ¹³Veterinary Wildlife Services, South African National Parks, 97 Memorial Road, Old Testing Grounds, Kimberley 8301, South Africa. ¹⁴Mammal Research Institute, Department of Zoology and Entomology, University of Pretoria, Pretoria 002, South Africa. ¹⁵Instituto Andino Patagónico de Tecnologías Biológicas y Geoambientales, Grupo GEA, IPATEC-UNCO-CONICET, San Carlos de Bariloche, Rio Negro, Argentina. ¹⁶Red Sea Research Centre, King Abdullah University of Science and Technology, Thuwal 23955, Saudi Arabia. ¹⁷Center for Zoo and Wild Animal Health, Copenhagen Zoo, Roskildevej 38, 2000 Frederiksberg,

Denmark. ¹⁸Department of Zoology and Entomology, University of Fort Hare Alice Campus, Ring Road, Alice 5700, South Africa.

Received: 8 March 2021 Accepted: 26 May 2021

Published online: 01 July 2021

References

- Browning E, Bolton M, Owen E, Shoji A, Guilford T, Freeman R. Predicting animal behaviour using deep learning: GPS data alone accurately predict diving in seabirds. *Methods Ecol Evol.* 2018;9(3):681–92.
- McClune DW. Joining the dots: reconstructing 3D environments and movement paths using animal-borne devices. *Anim Biotelem.* 2018;6(1):5.
- Schlägel UE, Signer J, Herde A, Eden S, Jeltsch F, Eccard JA, Dammhahn M. Estimating interactions between individuals from concurrent animal movements. *Methods Ecol Evol.* 2019;10(8):1234–45.
- Cagnacci F, Boitani L, Powell RA, Boyce MS. Animal ecology meets GPS-based radiotelemetry: a perfect storm of opportunities and challenges. *Philos Trans R Soc B Biol Sci.* 2010;365(1550):2157–62.
- Williams HJ, Taylor LA, Benhamou S, Bijleveld AI, Clay TA, de Grissac S, Demšar U, English HM, Franconi N, Gómez-Laich A. Optimizing the use of biologgers for movement ecology research. *J Anim Ecol.* 2020;89(1):186–206.
- Cotter CH. Early dead reckoning navigation. *J Navig.* 1978;31(1):20–8.
- Levi RW, Judd T. Dead reckoning navigational system using accelerometer to measure foot impacts. In: Point Research Corporation, Santa Meijer, et al., *Methods to assess physical activity with Ana, Calif.*, vol. 5,583,776. 1996. p. 8.
- Beauregard S, Haas H. Pedestrian dead reckoning: a basis for personal positioning. In: *Proceedings of the 3rd workshop on positioning, navigation and communication*; 2006. p. 27–35.
- Walker JS, Jones MW, Laramie RS, Holton MD, Shepard ELC, Williams HJ, Scantlebury DM, Marks NJ, Magowan EA, Maguire IE, Bidder OR, Di Virgilio A, Wilson RP. Prying into the intimate secrets of animal lives; software beyond hardware for comprehensive annotation in 'Daily Diary' tags. *Mov Ecol.* 2015;3(1):29.
- Bidder OR, Walker JS, Jones MW, Holton MD, Urge P, Scantlebury DM, Marks NJ, Magowan EA, Maguire IE, Wilson RP. Step by step: reconstruction of terrestrial animal movement paths by dead-reckoning. *Mov Ecol.* 2015;3(1):23.
- Wensveen PJ, Thomas L, Miller PJO. A path reconstruction method integrating dead-reckoning and position fixes applied to humpback whales. *Mov Ecol.* 2015;3(1):31.
- Andrzejczek S, Gleiss AC, Lear KO, Pattiaratchi CB, Chapple TK, Meekan MG. Biologging tags reveal links between fine-scale horizontal and vertical movement behaviors in Tiger Sharks (*Galeocerdo cuvier*). *Front Mar Sci.* 2019;6:229.
- Bowditch N. *The American practical navigator, Bicentennial edition.* Bethesda: National Imagery and Mapping Agency; 2002. p. 879.
- Wilson R, Wilson M-P. Dead reckoning: a new technique for determining penguin movements at sea. *Meeresforschung.* 1988;32(2):155–8.
- Wilson RP, Wilson M-P, Link R, Mempel H, Adams NJ. Determination of movements of African penguins *Spheniscus demersus* using a compass system: dead reckoning may be an alternative to telemetry. *J Exp Biol.* 1991;157(1):557–64.
- Wilson RP, Liebsch N, Davies IM, Quintana F, Weimerskirch H, Storch S, Lucke K, Siebert U, Zankl S, Müller G, Zimmer I, Scolaro A, Campagna C, Plötz J, Bornemann H, Teilmann J, McMahon CR. All at sea with animal tracks: methodological and analytical solutions for the resolution of movement. *Deep Sea Res II Top Stud Oceanogr.* 2007;54(3):193–210.
- Mitani Y, Watanabe Y, Sato K, Cameron MF, Naito Y. 3D diving behavior of Weddell seals with respect to prey accessibility and abundance. *Mar Ecol Prog Ser.* 2004;281:275–81.
- Elkaim GH, Decker EB, Oliver G, Wright B. Marine mammal marker (MAMMARK) dead reckoning sensor for In-Situ environmental monitoring. In: *Proceedings of IEEE/ION PLANS 2006, San Diego, CA*; 2006. p. 976–87.
- Wilson AD, Wikelski M, Wilson RP, Cooke SJ. Utility of biological sensor tags in animal conservation. *Conserv Biol.* 2015;29(4):1065–75.
- Wilson RP, Shepard E, Liebsch N. Prying into the intimate details of animal lives: use of a daily diary on animals. *Endanger Species Res.* 2008;4(1–2):123–37.
- Pedley M. eCompass-build and calibrate a tilt-compensating electronic compass. *Circuit Cellar Mag Comput Appl.* 2012;265:1–6.
- Li Z, Li X, Wang Y. A calibration method for magnetic sensors and accelerometer in tilt-compensated digital compass. In: *2009 9th international conference on electronic measurement & instruments*, 16–19 Aug 2009; 2009. p. 2-868-862-871.
- Ozyagcilar T. Implementing a tilt-compensated eCompass using accelerometer and magnetometer sensors. Freescale semiconductor. Application Note, AN4248; 2012.
- Gheorghie MV, Bodea MC. Calibration optimization study for tilt-compensated compasses. *IEEE Trans Instrum Meas.* 2018;67(6):1486–94.
- Liu Y, Battaile BC, Trites AW, Zidek JV. Bias correction and uncertainty characterization of dead-reckoned paths of marine mammals. *Anim Biotelem.* 2015;3(1):51.
- Dewhurst OP, Evans HK, Roskilly K, Harvey RJ, Hubel TY, Wilson AM. Improving the accuracy of estimates of animal path and travel distance using GPS drift-corrected dead reckoning. *Ecol Evol.* 2016;6(17):6210–22.
- Mitani Y, Sato K, Ito S, Cameron MF, Siniff DB, Naito Y. A method for reconstructing three-dimensional dive profiles of marine mammals using geomagnetic intensity data: results from two lactating Weddell seals. *Polar Biol.* 2003;26(5):311–7.
- Whitney NM, Pratt HL Jr, Pratt TC, Carrier JC. Identifying shark mating behaviour using three-dimensional acceleration loggers. *Endanger Species Res.* 2010;10:71–82.
- Lisovski S, Hewson CM, Klaassen RHG, Korner-Nievergelt F, Kristensen MW, Hahn S. Geolocation by light: accuracy and precision affected by environmental factors. *Methods Ecol Evol.* 2012;3(3):603–12.
- Miller PJO, Johnson MP, Madsen PT, Biassoni N, Quero M, Tyack PL. Using at-sea experiments to study the effects of airguns on the foraging behavior of sperm whales in the Gulf of Mexico. *Deep Sea Res I Oceanogr Res Pap.* 2009;56(7):1168–81.
- Baumgartner MF, Freitag L, Partan J, Ball KR, Prada KE. Tracking large marine predators in three dimensions: the real-time acoustic tracking system. *IEEE J Oceanic Eng.* 2008;33(2):146–57.
- Williams LR, Fox DR, Bishop-Hurley GJ, Swain DL. Use of radio frequency identification (RFID) technology to record grazing beef cattle water point use. *Comput Electron Agric.* 2019;156:193–202.
- Alexander JS, Zhang C, Shi K, Riordan P. A granular view of a snow leopard population using camera traps in Central China. *Biol Conserv.* 2016;197:27–31.
- English HM, Harvey L, Wilson RP, Gunner RM, Holton MD, Woodroffe R, Börger L. Multi-sensor biologgers and innovative training allow data collection with high conservation and welfare value in zoos. <https://doi.org/10.21203/rs.3.rs-562677/v1>.
- Fancy SG, Pank LF, Douglas DC, Curby CH, Garner GW. *Satellite telemetry: a new tool for wildlife research and management.* Washington, D.C: Fish and Wildlife Service Washington DC; 1988.
- Soutullo A, Cadahía L, Urios V, Ferrer M, Negro JJ. Accuracy of light-weight satellite telemetry: a case study in the Iberian Peninsula. *J Wildl Manag.* 2007;71(3):1010–5.
- Catipovic JA. Performance limitations in underwater acoustic telemetry. *IEEE J Oceanic Eng.* 1990;15(3):205–16.
- Hofman MPG, Hayward MW, Heim M, Marchand P, Rolandsen CM, Mattisson J, Urbano F, Heurich M, Mysterud A, Melzheimer J, Morellet N, Voigt U, Allen BL, Gehr B, Rouco C, Ullmann W, Holand Ø, Jørgensen NH, Steinheim G, Cagnacci F, Kroeschel M, Kaczensky P, Buuveibaatar B, Payne JC, Palmegiani I, Jerina K, Kjellander P, Johansson Ö, LaPoint S, Bayrakcismith R, Linnell JDC, Zaccaroni M, Jorge MLS, Oshima JEF, Songhurst A, Fischer C, Mc Bride RT Jr, Thompson JJ, Streif S, Sandfort R, Bonenfant C, Drouilly M, Klapproth M, Zinner D, Yarnell R, Stronza A, Wilcott L, Meisinger E, Thaker M, Vanak AT, Nicoloso S, Graeber R, Said S, Boudreau MR, Devlin A, Hoogsteijn R, May-Junior JA, Nifong JC, Odden J, Quigley HB, Tortato F, Parker DM, Caso A, Perrine J, Tellaecae C, Zieba F, Zwijacz-Kozica T, Appel CL, Axsom I, Bean WT, Cristescu B, Périquet S, Teichman KJ, Karpanty S, Licoppe A, Menges V, Black K, Scheppers TL, Schai-Braun SC, Azevedo FC, Lemos FG, Payne A, Swanepoel LH, Weckworth BV, Berger A, Bertassoni A, McCulloch G, Sustr P, Athreya V,

- Bockmuhl D, Casar J, Ekori A, Melovski D, Richard-Hansen C, van de Vyver D, Reyna-Hurtado R, Robardet E, Selva N, Sergiel A, Farhadinia MS, Sunde P, Portas R, Ambarli H, Berzins R, Kappeler PM, Mann GK, Pyritz L, Bissett C, Grant T, Steinmetz R, Swedell L, Welch RJ, Armenteras D, Bidder OR, González TM, Rosenblatt A, Kachel S, Balkenhol N. Right on track? Performance of satellite telemetry in terrestrial wildlife research. *PLoS ONE*. 2019;14(5):e0216223.
39. Newey S, Davidson P, Nazir S, Fairhurst G, Verdicchio F, Irvine RJ, van der Wal R. Limitations of recreational camera traps for wildlife management and conservation research: a practitioner's perspective. *Ambio*. 2015;44(4):624–35.
40. Leonardo M, Noss AJ, Erika C, Damián IR. Ocelot (*Felis pardalis*) population densities, activity, and ranging behaviour in the dry forests of eastern Bolivia: data from camera trapping. *J Trop Ecol*. 2005;21(3):349–53.
41. Lewis JS, Rachlow JL, Garton EO, Vierling LA. Effects of habitat on GPS collar performance: using data screening to reduce location error. *J Appl Ecol*. 2007;44(3):663–71.
42. Kaur M, Sandhu M, Mohan N, Sandhu PS. RFID technology principles, advantages, limitations & its applications. *Int J Comput Electr Eng*. 2011;3(1):151.
43. Davis RW, Fuiman LA, Williams TM, Le Boeuf BJ. Three-dimensional movements and swimming activity of a northern elephant seal. *Comp Biochem Physiol A Mol Integr Physiol*. 2001;129(4):759–70.
44. Wilson RP. Movements in Adélie Penguins foraging for chicks at Ardley Island, Antarctica: circles within spirals, wheels within wheels. *Polar Biosci*. 2002;15:75–87.
45. Johnson MP, Tyack PL. A digital acoustic recording tag for measuring the response of wild marine mammals to sound. *IEEE J Oceanic Eng*. 2003;28(1):3–12.
46. Shiomi K, Sato K, Mitamura H, Arai N, Naito Y, Ponganis PJ. Effect of ocean current on the dead-reckoning estimation of 3-D dive paths of emperor penguins. *Aquat Biol*. 2008;3(3):265–70.
47. Laplanche C, Marques TA, Thomas L. Tracking marine mammals in 3D using electronic tag data. *Methods Ecol Evol*. 2015;6(9):987–96.
48. Adachi T, Costa DP, Robinson PW, Peterson SH, Yamamichi M, Naito Y, Takahashi A. Searching for prey in a three-dimensional environment: hierarchical movements enhance foraging success in northern elephant seals. *Funct Ecol*. 2017;31(2):361–9.
49. Bras YL, Jouma'a J, Guinet C. Three-dimensional space use during the bottom phase of southern elephant seal dives. *Mov Ecol*. 2017;5(1):18.
50. Andrzejczek S, Gleiss AC, Pattiaratchi CB, Meekan MG. First Insights Into the fine-scale movements of the Sandbar Shark, *Carcharhinus plumbeus*. *Front Mar Sci*. 2018;5:483.
51. Aoki K, Amano M, Mori K, Kourugi A, Kubodera T, Miyazaki N. Active hunting by deep-diving sperm whales: 3D dive profiles and maneuvers during bursts of speed. *Mar Ecol Prog Ser*. 2012;444:289–301.
52. Narazaki T, Sato K, Abernathy K, Marshall G, Miyazaki N. Sea turtles compensate deflection of heading at the sea surface during directional travel. *J Exp Biol*. 2009;212(24):4019–26.
53. Benoit-Bird KJ, Battaile BC, Nordstrom CA, Trites AW. Foraging behavior of northern fur seals closely matches the hierarchical patch scales of prey. *Mar Ecol Prog Ser*. 2013;479:283–302.
54. Ware C, Friedlaender AS, Nowacek DP. Shallow and deep lunge feeding of humpback whales in fjords of the West Antarctic Peninsula. *Mar Mamm Sci*. 2011;27(3):587–605.
55. Wright BM, Ford JK, Ellis GM, Deecke VB, Shapiro AD, Battaile BC, Trites AW. Fine-scale foraging movements by fish-eating killer whales (*Orcinus orca*) relate to the vertical distributions and escape responses of salmonid prey (*Oncorhynchus* spp.). *Mov Ecol*. 2017;5(1):1–18.
56. Wensveen PJ, Isojunno S, Hansen RR, von Benda-Beckmann AM, Kleivane L, van IJsselmuide S, Lam FPA, Kvadsheim PH, DeRuiter SL, Curé C, Narazaki T, Tyack PL, Miller PJO. Northern bottlenose whales in a pristine environment respond strongly to close and distant navy sonar signals. *Proc R Soc B Biol Sci*. 1899;2019(286):20182592.
57. Ropert-Coudert Y, Wilson RP. Trends and perspectives in animal-attached remote sensing. *Front Ecol Environ*. 2005;3(8):437–44.
58. Francisco FA, Nührenberg P, Jordan A. High-resolution, non-invasive animal tracking and reconstruction of local environment in aquatic ecosystems. *Mov Ecol*. 2020;8(1):27.
59. Goldbogen JA, Calambokidis J, Shadwick RE, Oleson EM, McDonald MA, Hildebrand JA. Kinematics of foraging dives and lunge-feeding in fin whales. *J Exp Biol*. 2006;209(7):1231–44.
60. Zimmer WM, Tyack PL, Johnson MP, Hadsen PT. Three-dimensional beam pattern of regular sperm whale clicks confirms bent-horn hypothesis. *J Acoust Soc Am*. 2005;117(3):1473–85.
61. Wilson RP, Ropert-Coudert Y, Kato A. Rush and grab strategies in foraging marine endotherms: the case for haste in penguins. *Anim Behav*. 2002;63(1):85–95.
62. Gabaldon J, Turner EL, Johnson-Roberson M, Barton K, Johnson M, Anderson EJ, Shorter KA. Integration, calibration, and experimental verification of a speed sensor for swimming animals. *IEEE Sens J*. 2019;19(10):3616–25.
63. Denny M. Air and water: the biology and physics of life's media. Princeton University Press; 1993.
64. Altnayn K, Khan Mohammed A, Marengo M, Swanepoel L, Przybysz A, Muller C, Fahlman A, Buttner U, Gerdali NR, Wilson RP, Duarte CM, Kosel J. Wearable multifunctional printed graphene sensors. *NPJ Flex Electron*. 2019;3(1):1–10.
65. Williams H, Shepard E, Duriez O, Lambertucci SA. Can accelerometry be used to distinguish between flight types in soaring birds? *Anim Biotelem*. 2015;3(1):1–11.
66. Williams HJ, Shepard E, Holton MD, Alarcón P, Wilson R, Lambertucci S. Physical limits of flight performance in the heaviest soaring bird. *Proc Natl Acad Sci*. 2020;117(30):17884–90.
67. Wilson RP, Börger L, Holton MD, Scantlebury DM, Gómez-Laich A, Quintana F, Rosell F, Graf PM, Williams H, Gunner R, Hopkins L, Marks N, Gerdali NR, Duarte CM, Scott R, Strano MS, Robotka H, Eizaguirre C, Fahlman A, Shepard ELC. Estimates for energy expenditure in free-living animals using acceleration proxies: a reappraisal. *J Anim Ecol*. 2020;89(1):161–72.
68. Bidder OR, Qasem LA, Wilson RP. On higher ground: how well can dynamic body acceleration determine speed in variable terrain? *PLoS ONE*. 2012;7(11):e50556.
69. Bidder OR, Soresina M, Shepard ELC, Halsey LG, Quintana F, Gómez-Laich A, Wilson RP. The need for speed: testing acceleration for estimating animal travel rates in terrestrial dead-reckoning systems. *Zoology*. 2012;115(1):58–64.
70. Markovska M, Svensson R. Evaluation of drift correction strategies for an inertial based dairy cow positioning system: a study on tracking the position of dairy cows using a foot mounted IMU with drift correction from ZUPT or sparse RFID locations. Student thesis. 2019.
71. di Virgilio A, Morales JM, Lambertucci SA, Shepard EL, Wilson RP. Multi-dimensional precision livestock farming: a potential toolbox for sustainable rangeland management. *PeerJ*. 2018;6:e4867.
72. Shepard EL, Wilson RP, Halsey LG, Quintana F, Laich AG, Gleiss AC, Liebisch N, Myers AE, Norman B. Derivation of body motion via appropriate smoothing of acceleration data. *Aquat Biol*. 2008;4(3):235–41.
73. Noda T, Okuyama J, Koizumi T, Arai N, Kobayashi M. Monitoring attitude and dynamic acceleration of free-moving aquatic animals using a gyroscope. *Aquat Biol*. 2012;16(3):265–76.
74. Wilson JW, Mills MG, Wilson RP, Peters G, Mills ME, Speakman JR, Durant SM, Bennett NC, Marks NJ, Scantlebury M. Cheetahs, *Acinonyx jubatus*, balance turn capacity with pace when chasing prey. *Biol Lett*. 2013;9(5):20130620.
75. McNarry MA, Wilson RP, Holton MD, Griffiths IW, Mackintosh KA. Investigating the relationship between energy expenditure, walking speed and angle of turning in humans. *PLoS ONE*. 2017;12(8):e0182333.
76. Kaniewski P, Kazubek J. Integrated system for heading determination. *Acta Phys Pol Ser A Gen Phys*. 2009;116(3):325.
77. Noda T, Kawabata Y, Arai N, Mitamura H, Watanabe S. Animal-mounted gyroscope/accelerometer/magnetometer: in situ measurement of the movement performance of fast-start behaviour in fish. *J Exp Mar Biol Ecol*. 2014;451:55–68.
78. Patonis P, Patias P, Tziavos IN, Rossikopoulos D, Margaritis KG. A fusion method for combining low-cost IMU/magnetometer outputs for use in applications on mobile devices. *Sensors*. 2018;18(8):2616.
79. Diebel J. Representing attitude: Euler angles, unit quaternions, and rotation vectors. Technical Report, vol. 58. Stanford: Stanford University; 2006. p. 1–35.

80. Han S, Wang J. A novel method to integrate IMU and magnetometers in attitude and heading reference systems. *J Navig.* 2011;64(4):727–38.
81. Säll J, Merkel J. Indoor navigation using accelerometer and magnetometer. Student thesis. 2011.
82. Alaimo A, Artale V, Milazzo C, Ricciardello A. Comparison between Euler and quaternion parametrization in UAV dynamics. *AIP Conf Proc.* 2013;1558(1):1228–31.
83. Valenti RG, Dryanovski I, Xiao J. Keeping a good attitude: a quaternion-based orientation filter for IMUs and MARGs. *Sensors.* 2015;15(8):19302–30.
84. Feng K, Li J, Zhang X, Shen C, Bi Y, Zheng T, Liu J. A new quaternion-based Kalman filter for real-time attitude estimation using the two-step geometrically-intuitive correction algorithm. *Sensors.* 2017;17(9):2146.
85. Chiella ACB, Teixeira BOS, Pereira GAS. Quaternion-based robust attitude estimation using an adaptive unscented Kalman filter. *Sensors.* 2019;19(10):2372.
86. Gunner RM, Wilson RP, Holton MD, Scott R, Hopkins P, Duarte CM. A new direction for differentiating animal activity based on measuring angular velocity about the yaw axis. *Ecol Evol.* 2020;10(14):7872–86.
87. Wiltschko R. *Magnetic orientation in animals*, vol. 33. Berlin: Springer Science & Business Media; 2012.
88. Sequeira MM, Rickenbach M, Wietlisbach V, Tullen B, Schutz Y. Physical activity assessment using a pedometer and its comparison with a questionnaire in a large population survey. *Am J Epidemiol.* 1995;142(9):989–99.
89. Kerdok AE, Biewener AA, McMahon TA, Weyand PG, Herr HM. Energetics and mechanics of human running on surfaces of different stiffnesses. *J Appl Physiol.* 2002;92(2):469–78.
90. Halsey LG, Shepard ELC, Hulston CJ, Venables MC, White CR, Jeukendrup AE, Wilson RP. Acceleration versus heart rate for estimating energy expenditure and speed during locomotion in animals: tests with an easy model species, *Homo sapiens*. *Zoology.* 2008;111(3):231–41.
91. Miwa M, Oishi K, Nakagawa Y, Maeno H, Anzai H, Kumagai H, Okano K, Tobioka H, Hirooka H. Application of overall dynamic body acceleration as a proxy for estimating the energy expenditure of grazing farm animals: relationship with heart rate. *PLoS ONE.* 2015;10(6):e0128042.
92. Chapman Jason W, Klaassen Raymond HG, Drake VA, Fossette S, Hays Graeme C, Metcalfe Julian D, Reynolds Andrew M, Reynolds Don R, Alerstam T. Animal orientation strategies for movement in flows. *Curr Biol.* 2011;21(20):R861–70.
93. R Development Core Team. R—a language and environment for statistical computing. Vienna: R Foundation for Statistical Computing; 2021.
94. The R project for statistical computing. <https://www.r-project.org/>. Accessed 07 Mar 2021.
95. GundoG.Tracks GitHub database. Available at <https://github.com/Richard6195/Dead-reckoning-animal-movements-in-R>. Accessed 09 June 2021
96. Dowlle M, Srinivasan A, Gorecki J, Chirico M, Stetsenko P, Short T, Lianoglou S, Antonyan E, Bonsch M, Parsonage H. Package 'data.table'. Extension of 'data frame'; 2019.
97. Gunner RM, Wilson RP, Holton MD, Hopkins P, Bell SH, Marks NJ, Bennett NC, Ferreira S, Govender D, Viljoen P, Bruns A, Schalkwyk Lv, Bertelsen MF, Duarte CM, Rooyen MCv, Tambling CJ, Goppert A, Diesel D, Scantlebury DM. Decision rules for determining terrestrial movement and the consequences for filtering high-resolution GPS tracks—a case study using the African Lion (*Panthera leo*). *Mov Ecol.* (in review).
98. Dunford CE, Marks NJ, Wilmers CC, Bryce CM, Nickel B, Wolfe LL, Scantlebury DM, Williams TM. Surviving in steep terrain: a lab-to-field assessment of locomotor costs for wild mountain lions (*Puma concolor*). *Mov Ecol.* 2020;8(1):34.
99. Wilson RP, Rose KA, Gunner R, Holton M, Marks NJ, Bennett NC, Bell SH, Twining JP, Hesketh J, Duarte CM, Bezodis N, Scantlebury DM. Forces experienced by instrumented animals depend on lifestyle. *bioRxiv.* 2020. <https://doi.org/10.1101/2020.08.20.258756>.
100. Willener AST, Handrich Y, Halsey LG, Strike S. Effect of walking speed on the gait of king penguins: an accelerometric approach. *J Theor Biol.* 2015;387:166–73.
101. Wilson RP, Holton MD, di Virgilio A, Williams H, Shepard ELC, Lambertucci S, Quintana F, Sala JE, Balaji B, Lee ES, Srivastava M, Scantlebury DM, Duarte CM. Give the machine a hand: a Boolean time-based decision-tree template for rapidly finding animal behaviours in multi-sensor data. *Methods Ecol Evol.* 2018;9(11):2206–15.
102. Rong L, Zhiguo D, Jianzhong Z, Ming L. Identification of individual walking patterns using gait acceleration. In: 2007 1st international conference on bioinformatics and biomedical engineering, 6–8 July 2007; 2007. p. 543–6.
103. Wilson RP, Hustler K, Ryan PG, Burger AE, Noldeke EC. Diving birds in cold water: do Archimedes and Boyle determine energetic costs? *Am Nat.* 1992;140(2):179–200.
104. Aoki K, Watanabe YY, Crocker DE, Robinson PW, Biuw M, Costa DP, Miyazaki N, Fedak MA, Miller PJO. Northern elephant seals adjust gliding and stroking patterns with changes in buoyancy: validation of at-sea metrics of body density. *J Exp Biol.* 2011;214(17):2973–87.
105. Williams HJ, Holton MD, Shepard ELC, Largey N, Norman B, Ryan PG, Duriez O, Scantlebury M, Quintana F, Magowan EA, Marks NJ, Alagaili AN, Bennett NC, Wilson RP. Identification of animal movement patterns using tri-axial magnetometry. *Mov Ecol.* 2017;5(1):6.
106. Whitford M, Klimley AP. An overview of behavioral, physiological, and environmental sensors used in animal biotelemetry and biologging studies. *Anim Biotelem.* 2019;7(1):1–24.
107. Shamoun-Baranes J, Bom R, van Loon EE, Ens BJ, Oosterbeek K, Bouten W. From sensor data to animal behaviour: an oystercatcher example. *PLoS ONE.* 2012;7(5):e37997.
108. Wilson R. The Jackass Penguin (*Spheniscus demersus*) as a pelagic predator. *Mar Ecol Prog Ser.* 1985;25(3):219–27.
109. Culik BM, Wilson RP, Dannfeld R, Adelung D, Spairani HJ, Coria NRC. Pycgocelid penguins in a swim canal. *Polar Biol.* 1991;11(4):277–82.
110. Bethge P, Nicol S, Culik BM, Wilson RP. Diving behaviour and energetics in breeding little penguins (*Eudyptula minor*). *J Zool.* 1997;242(3):483–502.
111. Tobalske B, Dial K. Flight kinematics of black-billed magpies and pigeons over a wide range of speeds. *J Exp Biol.* 1996;199(2):263–80.
112. Sato K, Sakamoto KQ, Watanuki Y, Takahashi A, Katsumata N, Bost C-A, Weimerskirch H. Scaling of soaring seabirds and implications for flight abilities of giant pterosaurs. *PLoS ONE.* 2009;4(4):e5400.
113. Scharold J, Lai NC, Lowell W, Graham J. Metabolic rate, heart rate, and tailbeat frequency during sustained swimming in the leopard shark *Triakis semifasciata*. *Exp Biol.* 1989;48(4):223–30.
114. Kawabe R, Kawano T, Nakano N, Yamashita N, Hiraiishi T, Naito Y. Simultaneous measurement of swimming speed and tail beat activity of free-swimming rainbow trout *Oncorhynchus mykiss* using an acceleration data-logger. *Fish Sci.* 2003;69(5):959–65.
115. Steinhilber MF, Steffensen JF, Andersen NG. Tail beat frequency as a predictor of swimming speed and oxygen consumption of saithe (*Pollachius virens*) and whiting (*Merlangius merlangus*) during forced swimming. *Mar Biol.* 2005;148(1):197–204.
116. Miller PJO, Johnson MP, Tyack PL, Terray EA. Swimming gaits, passive drag and buoyancy of diving sperm whales *Physeter macrocephalus*. *J Exp Biol.* 2004;207(11):1953–67.
117. Laich AG, Wilson RP, Quintana F, Shepard EL. Identification of imperial cormorant *Phalacrocorax atriceps* behaviour using accelerometers. *Endanger Species Res.* 2008;10:29–37.
118. Chakravarty P, Cozzi G, Ozgul A, Aminian K. A novel biomechanical approach for animal behaviour recognition using accelerometers. *Methods Ecol Evol.* 2019;10(6):802–14.
119. Chakravarty P, Maalberg M, Cozzi G, Ozgul A, Aminian K. Behavioural compass: animal behaviour recognition using magnetometers. *Mov Ecol.* 2019;7(1):28.
120. Hochscheid S. Why we mind sea turtles' underwater business: a review on the study of diving behavior. *J Exp Mar Biol Ecol.* 2014;450:118–36.
121. Constandache I, Bao X, Azizyan M, Choudhury RR. Did you see Bob? human localization using mobile phones. In: Proceedings of the sixteenth annual international conference on Mobile computing and networking; Chicago, Illinois, USA. Association for Computing Machinery; 2010. p. 149–60.
122. Symington A, Trigoni N. Encounter based sensor tracking. In: Proceedings of the thirteenth ACM international symposium on mobile ad hoc

- networking and computing, Hilton Head, South Carolina, USA. Association for Computing Machinery; 2012. p. 15–24.
123. Harja YD, Sarno R. Determine the best option for nearest medical services using Google maps API, Haversine and TOPSIS algorithm. In: 2018 international conference on information and communications technology (ICOIACT), 6–7 March 2018; 2018. p. 814–9.
 124. Great circle distances and bearings between two locations. https://dtcenter.org/met/users/docs/write_ups/gc_simple.pdf.
 125. Robusto CC. The Cosine-Haversine formula. *Am Math Mon*. 1957;64(1):38–40.
 126. Zeileis A, Grothendieck G, Ryan JA, Andrews F, Zeileis MA. Package ‘zoo’.
 127. Jockers ML, Thalken R. Introduction to dplyr. In: Text analysis with r for students of literature. Cham: Springer; 2020. p. 121–32.
 128. Calculating Azimuth, distance, and altitude from a pair of GPS locations in JavaScript. <https://medium.com/javascript-in-plain-english/calculating-azimuth-distance-and-altitude-from-a-pair-of-gps-locations-36b4325d8ab0>.
 129. Handcock RN, Swain DL, Bishop-Hurley GJ, Patison KP, Wark T, Valencia P, Corke P, O'Neill CJ. Monitoring Animal behaviour and environmental interactions using wireless sensor networks, GPS collars and satellite remote sensing. *Sensors*. 2009;9(5):3586–603.
 130. Gužvica G, Bošnjak I, Bielen A, Babić D, Radanović-Gužvica B, Šver L. Comparative analysis of three different methods for monitoring the use of green bridges by wildlife. *PLoS ONE*. 2014;9(8):e106194.
 131. Patel A, Stocks B, Fisher C, Nicolls F, Boje E. Tracking the cheetah tail using animal-borne cameras, GPS, and an IMU. *IEEE Sens Lett*. 2017;1(4):1–4.
 132. Latham ADM, Latham MC, Anderson DP, Cruz J, Herries D, Hebblewhite M. The GPS craze: six questions to address before deciding to deploy GPS technology on wildlife. *N Z J Ecol*. 2015;39(1):143–52.
 133. Hebblewhite M, Haydon DT. Distinguishing technology from biology: a critical review of the use of GPS telemetry data in ecology. *Philos Trans R Soc B Biol Sci*. 2010;365(1550):2303–12.
 134. Gibb R, Shoji A, Fayet AL, Perrins CM, Guilford T, Freeman R. Remotely sensed wind speed predicts soaring behaviour in a wide-ranging pelagic seabird. *J R Soc Interface*. 2017;14(132):20170262.
 135. Swain DL, Wark T, Bishop-Hurley GJ. Using high fix rate GPS data to determine the relationships between fix rate, prediction errors and patch selection. *Ecol Model*. 2008;212(3):273–9.
 136. Ryan PG, Petersen SL, Peters G, Grémillet D. GPS tracking a marine predator: the effects of precision, resolution and sampling rate on foraging tracks of African Penguins. *Mar Biol*. 2004;145(2):215–23.
 137. Bouvet D, Garcia G. GPS latency identification by Kalman filtering. *Robotica*. 2000;18(5):475–85.
 138. Frair JL, Fieberg J, Hebblewhite M, Cagnacci F, DeCesare NJ, Pedrotti L. Resolving issues of imprecise and habitat-biased locations in ecological analyses using GPS telemetry data. *Philos Trans R Soc B Biol Sci*. 2010;365(1550):2187–200.
 139. Péron G, Calabrese JM, Duriez O, Fleming CH, García-Jiménez R, Johnston A, Lambertucci SA, Safi K, Shepard ELC. The challenges of estimating the distribution of flight heights from telemetry or altimetry data. *Anim Biotelem*. 2020;8(1):5.
 140. Poulin M-P, Clermont J, Berteaux D. Extensive daily movement rates measured in territorial arctic foxes. *Ecol Evol*. 2021;00:1–12.
 141. Marcus Rowcliffe J, Carbone C, Kays R, Kranstauber B, Jansen PA. Bias in estimating animal travel distance: the effect of sampling frequency. *Methods Ecol Evol*. 2012;3(4):653–62.
 142. Belant JL. Effects of antenna orientation and vegetation on global positioning system telemetry collar performance. *Northeast Nat*. 2009;16(4):577–84, 578.
 143. Graf PM, Wilson RP, Qasem L, Hackländer K, Rosell F. The use of acceleration to code for animal behaviours; a case study in free-ranging Eurasian beavers castor fiber. *PLoS ONE*. 2015;10(8):e0136751.
 144. Tonini MH, Palma ED. Tidal dynamics on the North Patagonian Argentinean Gulfs. *Estuar Coast Shelf Sci*. 2017;189:115–30.
 145. Wilson RP, Kreye JM, Lucke K, Urquhart H. Antennae on transmitters on penguins: balancing energy budgets on the high wire. *J Exp Biol*. 2004;207(15):2649–62.
 146. Wilson R, Griffiths I, Legg P, Friswell M, Bidder O, Halsey L, Lambertucci SA, Shepard E. Turn costs change the value of animal search paths. *Ecol Lett*. 2013;16(9):1145–50.
 147. Potts J, Börger L, Scantlebury DM, Bennett NC, Alagaili A, Wilson RP. Finding turning-points in ultra-high-resolution animal movement data. *Methods Ecol Evol*. 2018;9(10):2091–101.
 148. Narazaki T, Nakamura I, Aoki K, Iwata T, Shiomi K, Luschi P, Suganuma H, Meyer CG, Matsumoto R, Bost CA, Handrich Y, Amano M, Okamoto R, Mori K, Ciccione S, Bourjea J, Sato K. Similar circling movements observed across marine megafauna taxa. *iScience*. 2021;24:102221.
 149. Kunčar A, Sysel M, Urbánek T. Calibration of low-cost triaxial magnetometer. In: MATEC web of conferences 20th international conference on circuits, systems, communications and computers (CSCC 2016), 2016. EDP Sciences.
 150. Sodhi R, Prunty J, Hsu G, Oh B. Automatic calibration of a three-axis magnetic compass. U.S. Patent 7,451,549 PNI Corporation (Santa Rosa, CA, US); 2008.
 151. Renaudin V, Afzal MH, Lachapelle G. Complete triaxis magnetometer calibration in the magnetic domain. *J Sens*. 2010;2010:967245.
 152. Tabatabaei SAH, Gluhak A, Tafazolli R. A fast calibration method for triaxial magnetometers. *IEEE Trans Instrum Meas*. 2013;62(11):2929–37.
 153. Vitali A. Ellipsoid or sphere fitting for sensor calibration, Dt0059. ST Microelectronics, Design Tip; 2016.
 154. Simple and effective magnetometer calibration. <https://github.com/kriswiner/MPU6050/wiki/Simple-and-Effective-Magnetometer-Calibration>.
 155. Premerlani W, Bizard P. Direction cosine matrix imu: theory. *Diy Drone: Usa*; 2009:13–15.
 156. Grygorenko V. Sensing-magnetic compass with tilt compensation. Cypress perform, semiconductor. Application Notes, AN2272. 2011.
 157. Gleiss AC, Wilson RP, Shepard ELC. Making overall dynamic body acceleration work: on the theory of acceleration as a proxy for energy expenditure. *Methods Ecol Evol*. 2011;2(1):23–33.
 158. Choi S, Youn I-H, LeMay R, Burns S, Youn J-H. Biometric gait recognition based on wireless acceleration sensor using k-nearest neighbor classification. In: 2014 international conference on computing, networking and communications (ICNC), 3–6 Feb 2014; 2014. p. 1091–5.
 159. Sato K, Mitani Y, Cameron MF, Siniff DB, Naito Y. Factors affecting stroking patterns and body angle in diving Weddell seals under natural conditions. *J Exp Biol*. 2003;206(9):1461–70.
 160. Li W, Wang J. Effective adaptive Kalman filter for MEMS-IMU/magnetometers integrated attitude and heading reference systems. *J Navig*. 2013;66(1):99–113.
 161. Pedley M. Tilt sensing using a three-axis accelerometer. Freescale semiconductor. Application Note 2461. 2013;1 (Rev. 6):1–22.
 162. Shiomi K, Narazaki T, Sato K, Shimatani K, Arai N, Ponganis PJ, Miyazaki N. Data-processing artefacts in three-dimensional dive path reconstruction from geomagnetic and acceleration data. *Aquat Biol*. 2010;8(3):299–304.
 163. Evans P. Rotations and rotation matrices. *Acta Crystallogr Sect D*. 2001;57(10):1355–9.
 164. Wilson RP, Holton MD, Walker JS, Shepard ELC, Scantlebury DM, Wilson VL, Wilson GI, Tysse B, Gravenor M, Ciancio J, McNarry MA, Mackintosh KA, Qasem L, Rosell F, Graf PM, Quintana F, Gomez-Laich A, Sala J-E, Mulvenna CC, Marks NJ, Jones MW. A spherical-plot solution to linking acceleration metrics with animal performance, state, behaviour and lifestyle. *Mov Ecol*. 2016;4(1):22.
 165. Caruso MJ. Applications of magnetic sensors for low cost compass systems. In: IEEE 2000 position location and navigation symposium (Cat No00CH37062): 13–16 March 2000; 2000. p. 177–84.
 166. Boelter KDH. Aircraft magnetic declinator system, vol. 2020/0182619. Chicago: The Boeing Company; 2020.
 167. World magnetic model 2020 calculator. http://www.geomag.bgs.ac.uk/data_service/models_compass/wmm_calc.html. Accessed 07 Mar 2021.
 168. Qasem L, Cardew A, Wilson A, Griffiths I, Halsey LG, Shepard ELC, Gleiss AC, Wilson R. Tri-axial dynamic acceleration as a proxy for animal energy expenditure; should we be summing values or calculating the vector? *PLoS ONE*. 2012;7(2):e31187.

169. Pewsey A, Neuhäuser M, Ruxton GD. Circular statistics in R. OUP Oxford; 2013.
170. Farrahi V, Niemelä M, Kangas M, Korpelainen R, Jämsä T. Calibration and validation of accelerometer-based activity monitors: a systematic review of machine-learning approaches. *Gait Posture*. 2019;68:285–99.
171. Ropert-Coudert Y, Kato A, Baudat J, Bost C-A, Le Maho Y, Naito Y. Time/depth usage of Adélie penguins: an approach based on dive angles. *Polar Biol*. 2001;24(6):467–70.
172. Movable type scripts. <https://www.movable-type.co.uk/scripts/latlong.html>. Accessed 07 Mar 2021.
173. Grolemond G, Wickham H. Dates and times made easy with lubridate. *J Stat Softw*. 2011;40(i03):1–25.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Ready to submit your research? Choose BMC and benefit from:

- fast, convenient online submission
- thorough peer review by experienced researchers in your field
- rapid publication on acceptance
- support for research data, including large and complex data types
- gold Open Access which fosters wider collaboration and increased citations
- maximum visibility for your research: over 100M website views per year

At BMC, research is always in progress.

Learn more biomedcentral.com/submissions

