# Package 'imuf'

February 12, 2025

**Title** Estimate Orientation of an Inertial Measurement Unit

**Version** 0.6.0

**Description** Estimate the orientation of an inertial measurement unit (IMU) with a 3-axis accelerometer and a 3-axis gyroscope using a complementary filter. 'imuf' takes an IMU's accelerometer and gyroscope readings, time duration, its initial orientation, and a gain factor as inputs, and returns an estimate of the IMU's final orientation.

**License** MIT + file LICENSE

**Encoding** UTF-8

**RoxygenNote** 7.3.2

**LinkingTo** Rcpp, RcppEigen

**Imports** htmltools, htmlwidgets, Rcpp

**Suggests** knitr, rmarkdown, testthat (>= 3.0.0), purrr, ggplot2, shiny, serial, stringr

**Config/testthat/edition** 3

**URL** https://github.com/gitboosting/imuf, https://gitboosting.github.io/imuf/

**BugReports** https://github.com/gitboosting/imuf/issues

**Depends** R (>= 2.10)

**LazyData** true

**VignetteBuilder** knitr

**NeedsCompilation** yes

**Author** Felix Chan [aut, cre, cph], Ricardo Cabello [cph] ('three.js' JavaScript library)

**Maintainer** Felix Chan <chanfelix@gmail.com>

**Repository** CRAN

**Date/Publication** 2025-02-12 15:30:06 UTC

# Contents



---

`animate_imu` *Animate movement of an inertial measurement unit*

---

## Description

Create interactive animated movements of an inertial measurement unit

## Usage

```
animate_imu(data, dt, width = NULL, height = NULL, elementId = NULL)
```

## Arguments

| | |
|---|---|
| `data` | A `list` of numeric 4-vectors each of which a unit quaternion (w,x,y,z) |
| `dt` | A numeric of time duration in milli-seconds |
| `width, height` | width and height of animate_imu htmlwidget specified in any valid `CSS` size unit |
| `elementId` | `string` id as a valid `CSS` element id. |

---

`animate_imu-shiny` *Shiny bindings for animate_imu*

---

## Description

Output and render functions for using animate_imu within Shiny applications and interactive Rmd documents.

## Usage

```
animate_imuOutput(outputId, width = "100%", height = "400px")

renderAnimate_imu(expr, env = parent.frame(), quoted = FALSE)
```

### Arguments

| | |
|---|---|
| outputId | output variable to read from |
| width, height | Must be a valid CSS unit (like '100%', '400px', 'auto') or a number, which will be coerced to a string and have 'px' appended. |
| expr | An expression that generates a animate_imu |
| env | The environment in which to evaluate expr. |
| quoted | Is expr a quoted expression (with quote())? This is useful if you want to save an expression in a variable. |

---

## compUpdate *Update orientation with 3-axis acc and gyr data*

---

### Description

compUpdate() uses complementary filtering to update the orientation, given an initial orientation, readings of a 3-axis accelerometer and a 3-axis gyroscope. time duration, and a gain factor

### Usage

```
compUpdate(acc, gyr, dt, initQuat, gain)
```

### Arguments

| | |
|---|---|
| acc | A numeric 3-vector of 3-axis accelerometer readings in g |
| gyr | A numeric 3-vector of 3-axis gyroscope readings in rad/sec |
| dt | A numeric of time duration in sec |
| initQuat | A numeric 4-vector of the starting orientation in quaternion |
| gain | A numeric gain factor between 0 and 1 |

### Value

A numeric 4-vector of the ending orientation in quaternion

### Examples

```
compUpdate(c(0, 0, -1), c(1, 0, 0), 0.1, c(1, 0, 0, 0), 0.1)
```

`imu_object` *Animate in real time movement of an inertial measurement unit in shiny*

## Description

Create in real time animated movements of an inertial measurement unit in shiny

## Usage

```
imu_object(data = c(1, 0, 0, 0), width = NULL, height = NULL, elementId = NULL)
```

## Arguments

| | |
|---|---|
| `data` | A `vector` of numeric 4-vectors each of which a unit quaternion (w,x,y,z) |
| `width, height` | width and height of imu_object htmlwidget specified in any valid `CSS` size unit |
| `elementId` | `string` id as a valid `CSS` element id. |

`imu_object-shiny` *Shiny bindings for imu_object*

## Description

Output and render functions for using imu_object within Shiny applications and interactive Rmd documents.

## Usage

```
imu_objectOutput(outputId, width = "100%", height = "400px")

renderImu_object(expr, env = parent.frame(), quoted = FALSE)
```

## Arguments

| | |
|---|---|
| `outputId` | output variable to read from |
| `width, height` | Must be a valid CSS unit (like `'100%'`, `'400px'`, `'auto'`) or a number, which will be coerced to a string and have `'px'` appended. |
| `expr` | An expression that generates a imu_object |
| `env` | The environment in which to evaluate `expr`. |
| `quoted` | Is `expr` a quoted expression (with `quote()`)? This is useful if you want to save an expression in a variable. |

## imu_object-update *Update a widget in shiny with an orientation*

### Description

Pair of functions used in conjunction to update a imu_object widget

### Usage

```
imu_proxy(id, session = shiny::getDefaultReactiveDomain())

imu_send_data(proxy, data)
```

### Arguments

| | |
|---|---|
| id | HTML element id that houses the html widget |
| session | shiny session - default to current session |
| proxy | imu_proxy output to be used as input to imu_send_data |
| data | a numeric unit 4-vector (quaternion) for the updated orientation |

## rotV *Rotate a 3-vector by a quaternion*

### Description

`rotV()` rotates a 3-vector by a quaternion expressed as a unit 4-vector in (w,x,y,z) convention

### Usage

```
rotV(quat, vin)
```

### Arguments

| | |
|---|---|
| quat | A numeric unit 4-vector (w,x,y,z) for a rotation quaternion |
| vin | A numeric 3-vector to be rotated by quat |

### Value

A numeric 3-vector after the rotation

### Examples

```
q <- c(cos(pi/4), sin(pi/4), 0, 0)
vin <- c(0, 1, 0)
rotV(q, vin)
```

walking_shin_1 *University of Mannheim Real World Activities*

## Description

Accelerations and angular velocities captured by sensors of the mobile device situated on the shin of subject 1 while the subject was walking for 10 minutes

## Usage

```
walking_shin_1
```

## Format

`walking_shin_1`**:**

A data frame with 31,946 rows and 6 columns of accelerometer and gyroscope measurements at 50 Hz

**acc_x, acc_y, acc_z** north, east and down acceleration in m/s^2

**gyr_x, gyr_y, gyr_z** angular velocity about north, east and down in rad/sec

## Source

https://www.uni-mannheim.de/dws/research/projects/activity-recognition/dataset/dataset-realworld/s1/

# Index