Shimmer IMU sensors

Ευαγγελία Αθανασακη

HARDWARE

- Shimmer Device:

- Sensors included in each device:
    1. Accelerometer (3D), measures changes in velocity
    2. Gyroscope (3D), measures angular velocity
    3. Magnetometer (3D), measures strength of the magnetic field

HARDWARE
ACCELEROMETER

Figure 1. Shimmer3 default axis directions

$$\underline{a} = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = \begin{bmatrix} a_I \cos(\theta_x) + g \cos(\varphi_x) \\ a_I \cos(\theta_y) + g \cos(\varphi_y) \\ a_I \cos(\theta_z) + g \cos(\varphi_z) \end{bmatrix}$$

Figure 2. total acceleration vector

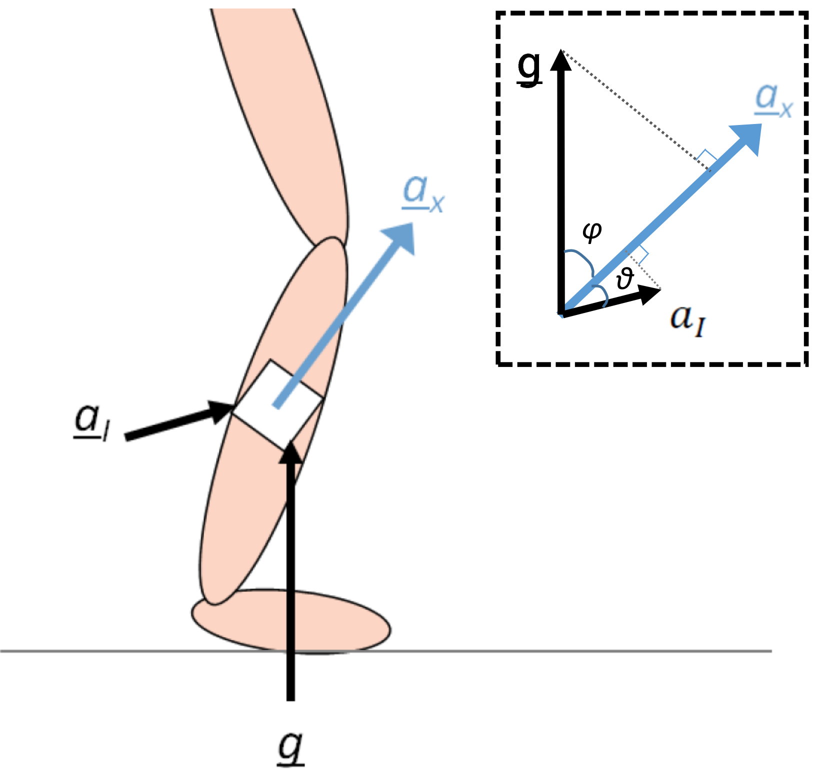


Figure 3. Uni-axial magnetometer attached to a leg segment

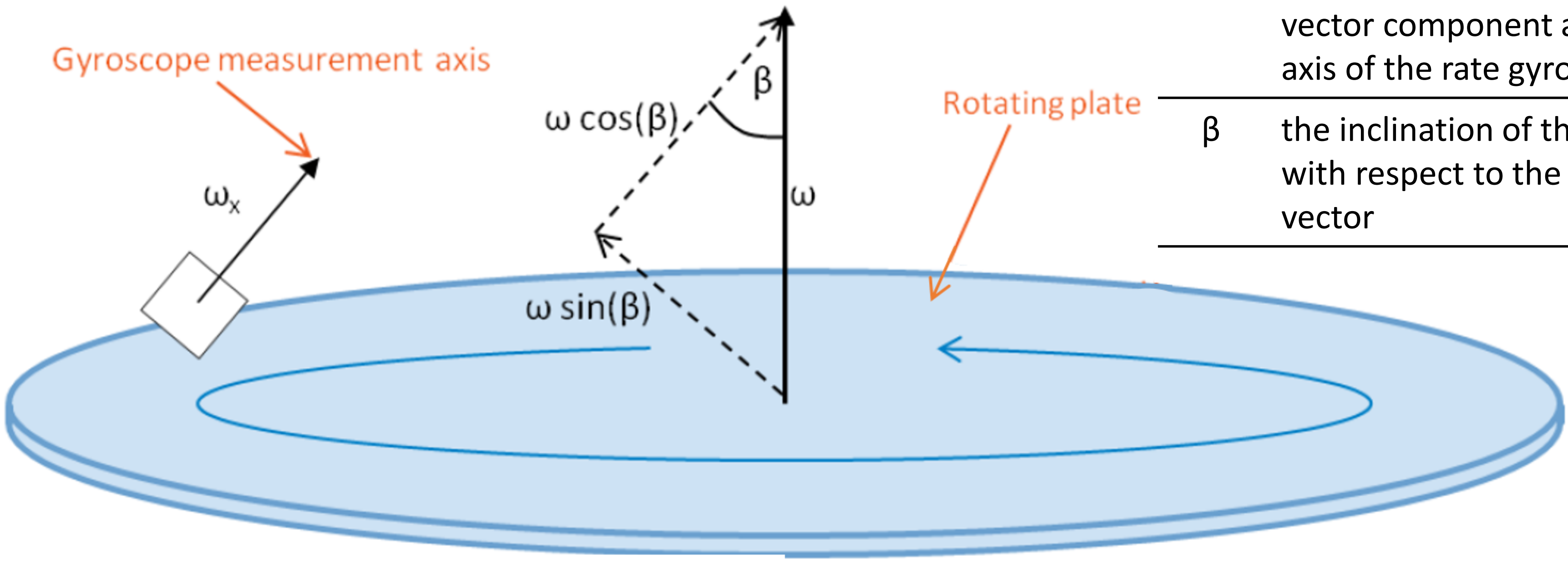
g	Gravitational acceleration, 1 g ≈ 9.81 m/s2
a	inertial acceleration, rate of change of velocity due to forces on body
θ	angle that the axis α_χ, makes with the inertial acceleration vector a_I
ϕ	angle that the axis α_χ, makes with the gravity vector g

HARDWARE
GYROSCOPE

$$\underline{\omega} = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} \omega \cos(\beta_x) \\ \omega \cos(\beta_y) \\ \omega \cos(\beta_z) \end{bmatrix}$$

Figure 2. tri-axial gyroscope rate

Figure 1. Shimmer3 default axis directions



ω	the magnitude of the angular velocity vector component along the measuring axis of the rate gyroscope
β	the inclination of the measuring axis with respect to the angular velocity vector

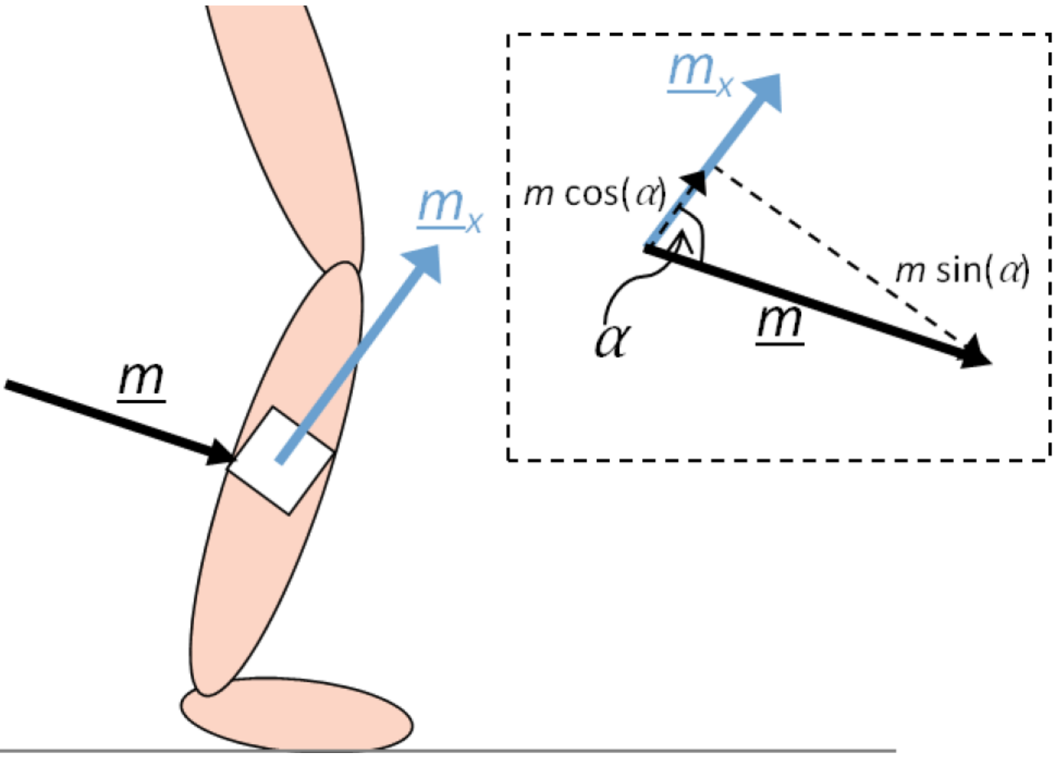
Figure 3. Uni-axial rate gyroscope attached to a rotating plate

HARDWARE
MAGNETOMETER

$$\underline{m} = \begin{bmatrix} m_x \\ m_y \\ m_z \end{bmatrix} = \begin{bmatrix} m\cos(\alpha_x) \\ m\cos(\alpha_y) \\ m\cos(\alpha_z) \end{bmatrix}$$

Figure 2. tri-axial magnetometer rate

Figure 1. Shimmer3 default axis directions



m	the magnitude of the magnetic field vector component along the measuring axis of the magnetometer
α	the angle between the magnetometer measuring axis and the magnetic field vector

SOFTWARE

The output $\mathbf{Y}$ for each sensor (acc/gyr/mag) is:

$$\underline{Y} = K r \underline{u} + \underline{b} + \underline{n}$$

$$\underline{Y} = \begin{bmatrix} Y_x \\ Y_y \\ Y_z \end{bmatrix} \qquad \underline{b} = \begin{bmatrix} b_x \\ b_y \\ b_z \end{bmatrix} \qquad \underline{n} = \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix} \qquad \underline{\upsilon} = \begin{bmatrix} \upsilon_x \\ \upsilon_y \\ \upsilon_z \end{bmatrix}$$

$$K = \begin{bmatrix} k_x & 0 & 0 \\ 0 & k_y & 0 \\ 0 & 0 & k_z \end{bmatrix} \qquad R = \begin{bmatrix} r_{x'x} & r_{x'y} & r_{x'z} \\ r_{y'x} & r_{y'y} & r_{y'z} \\ r_{z'x} & r_{z'y} & r_{z'z} \end{bmatrix}$$

K	Sensor’s scale factor (the change in output per unit of input Volts/g)	Estimated by calibration
r	is the rotation matrix which defines the actual sensor axes	Estimated by calibration
u	value of the sensed phenomenon	Defined in use
b	value of the sensor output when the sensed phenomenon is equal to zero	Estimated by calibration
n	is the noise vector	Defined by manufacturer

SOFTWARE

Data we actually get from each device have this format

TIMESTAMP (ms)	ACCELEROMETER (m/s^2)			GYROSCOPE (deg/sec)			MAGNETOMETER (local flux)		
	X axes	Y axes	Z axes	X axes	Y axes	Z axes	X axes	Y axes	Z axes
1024566650390	-0.6381260517	-0.85463565892	9.99691863372	-1.30603847151	0.77993576999	-1.71018442341	0.61133603238	-0.64372469635	0.72
1044097900390	-0.6380077624	-0.81825331906	10.0087475664	-1.06992021997	0.84110720293	-1.11508475673	0.61133603238	-0.64372469635	0.72
1063629150390	-0.5673273650	-0.79494647771	9.93393015842	-0.83719184135	0.91757149411	-0.85897238112	0.61133603238	-0.62348178137	0.72
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
3954254150390	0.21909503347	4.72064493121	8.81426530854	1.24207108972	1.66692154763	1.24207108972	0.61740890688	0.02834008097	0.81500000000
3973785400390	0.23075359958	4.72026912372	8.78964565140	1.79285476800	1.03991435999	1.79285476800	0.61740890688	0.02834008097	0.815000000001
3993316650390	0.27738063488	4.79104052609	8.69044442106	0.98309672126	0.55054289646	0.98309672126	0.61740890688	0.02834008097	0.81500000000

Units are in SI (International System of Units)

GESTURE IDENTIFICATION

Objective:

Identify four hand gestures

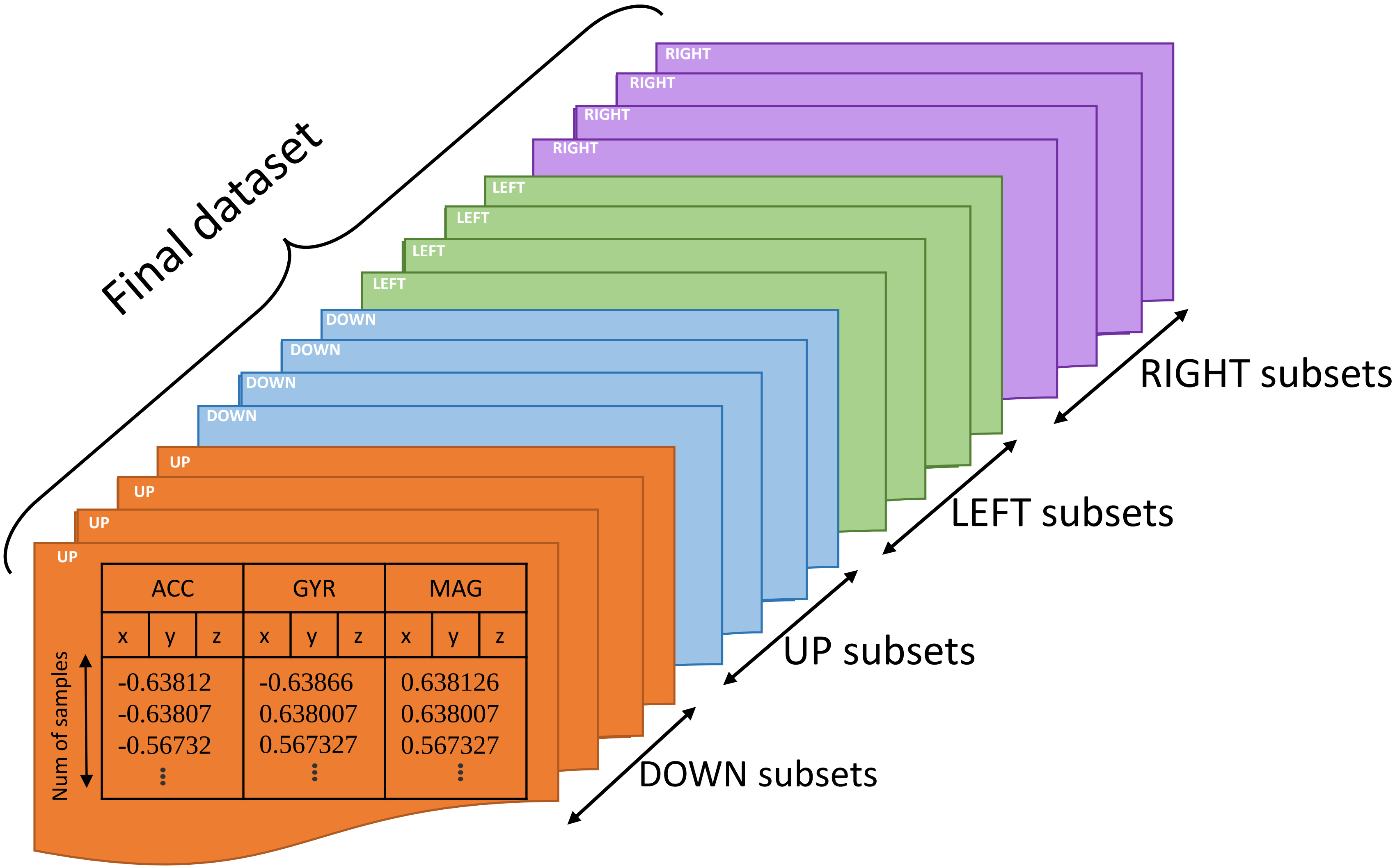
1. Up
2. Down
3. Left
4. Right

Methodology:

1. Construct final dataset by combining all gesture subsets
2. Extract features using statistical metrics
3. Apply k-means
4. Get results

1.DATASET

- Gather many repetitions of each gesture (each repetition is a subset)
- Compose all subsets and create final dataset



rms*: root mean square
stdev*: standard deviation

2.FEATURE EXTRACTION

For each subset from dataset, get mean/median/rms*/stdev* statistical metrics from all three sensors and for all dimensions of each

n-by-36 data matrix

- n: total number of subsets
- 36: features (3 dimensions/3 sensors/4 statistical metrics)

36 features

Total subsets

4 gestures

mean									rms									std									meadian								
acc			gyr			mag			acc			gyr			mag			acc			gyr			mag			acc			gyr			mag		
x	y	z	x	y	z	x	y	z	x	y	z	x	y	z	x	y	z	x	y	z	x	y	z	x	y	z	x	y	z	x	y	z	x	y	z

3.APPLY K-MEANS

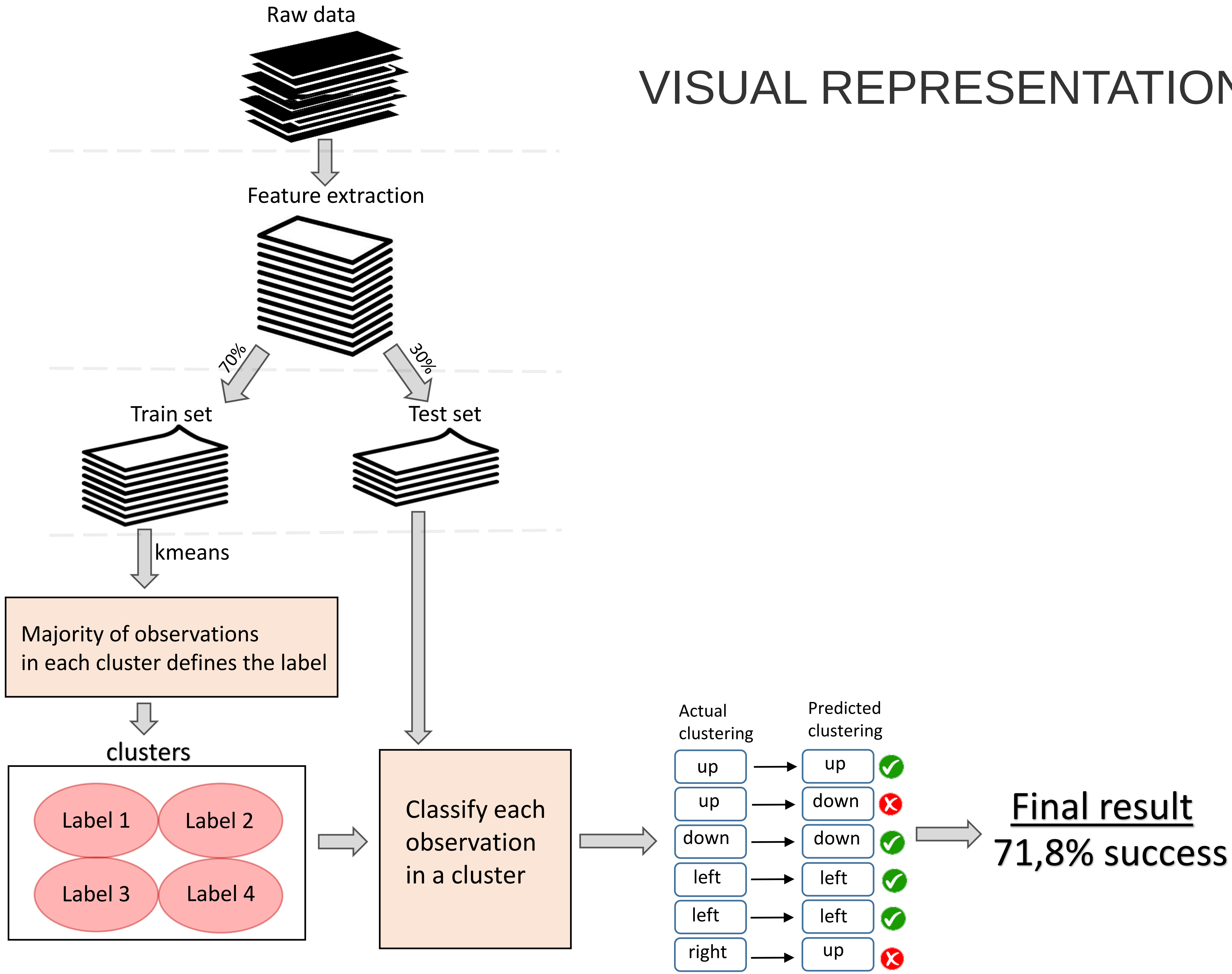
Parameters

- Data: n observations by 36 features matrix
- k=4 (clusters)
- Distance metric: cityblock

4.RESULTS

- K-fold for cross validation
- Get average success percentage

VISUAL REPRESENTATION



RESULTS

Size of dataset (subsets for each subject)	Controlled measurements	Left hand involved	Right hand involved	Fixed hand orientation per gesture	results
40	✓	✓	–	✓	96,87%
72	–	✓	✓	✓	89,29%
28	–	–	✓	–	29,28%

Controlled measurements: subject had total awareness about taking measurements
Fixed hand orientation per gesture: for all gesture repetitions hand had fixed orientation