Visual Analytics on Human Body Movement Data

Applied on Healthcare

Oky Purwantiningsih
Advisors: Arnaud Sallaberry, Jerômé Azé

Decision Support and Business Intelligence
IT4BI Master Thesis
Prepared at LIRMM and Université Paul Valéry Montpellier
In collaboration with NaturalPad

September 15, 2015

Outline

Motivation

(a) Kinect

(c) Wii Remote

(b) Wii Balance Board

(d) Play Station Move

Motion Sensing input devices enable players to control and interact with the game console through body movement.

Motivation

- Hammer and Planks is a serious game developed by NaturalPad, designed to train the equilibrium of patient with balance disorders.

Motivation

- Hammer and Planks is a serious game developed by NaturalPad, designed to train the equilibrium of patient with balance disorders.
- Players are required to move their body parts to the right, left, backward, and forward as part of the rehabilitation

Motivation

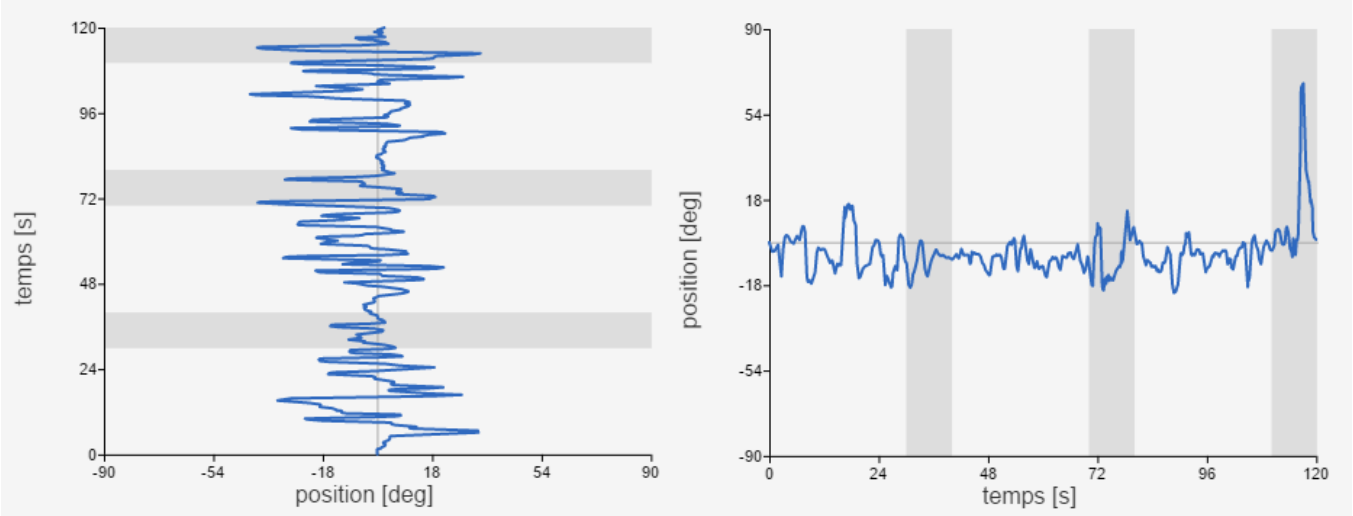
- Hammer and Planks is a serious game developed by NaturalPad, designed to train the equilibrium of patient with balance disorders.
- Players are required to move their body parts to the right, left, backward, and forward as part of the rehabilitation

Motivation

- Hammer and Planks is a serious game developed by NaturalPad, designed to train the equilibrium of patient with balance disorders.
- Players are required to move their body parts to the right, left, backward, and forward as part of the rehabilitation

- Game missions: collect bonuses, kill enemies, avoid obstacles

Motivation

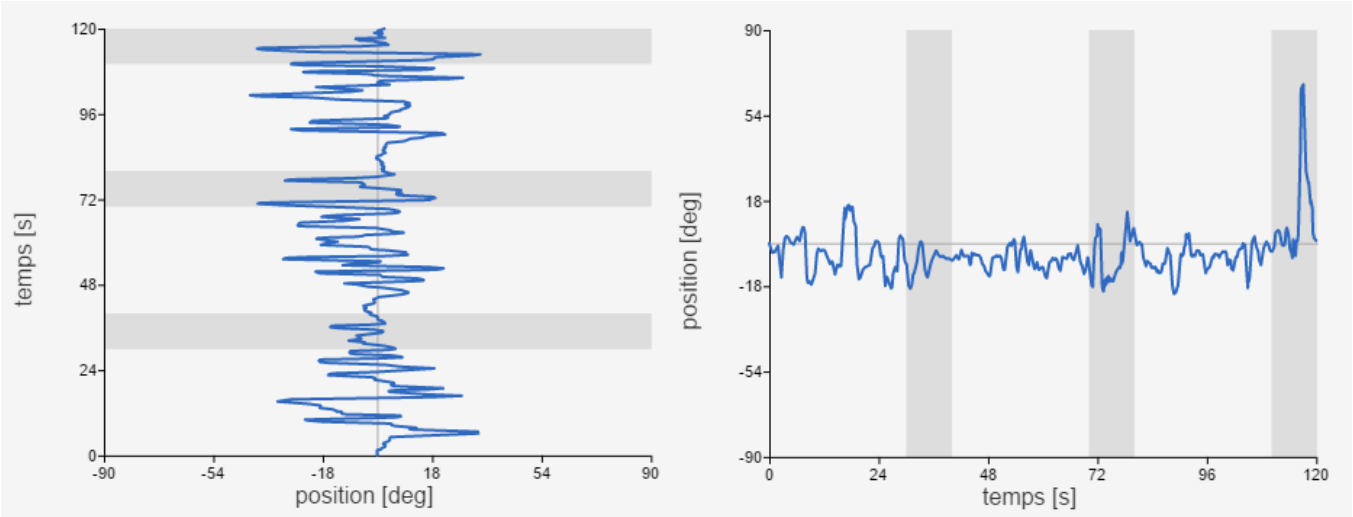


Current Visualization

Available:

- Average degree of body movement over x axis

Motivation

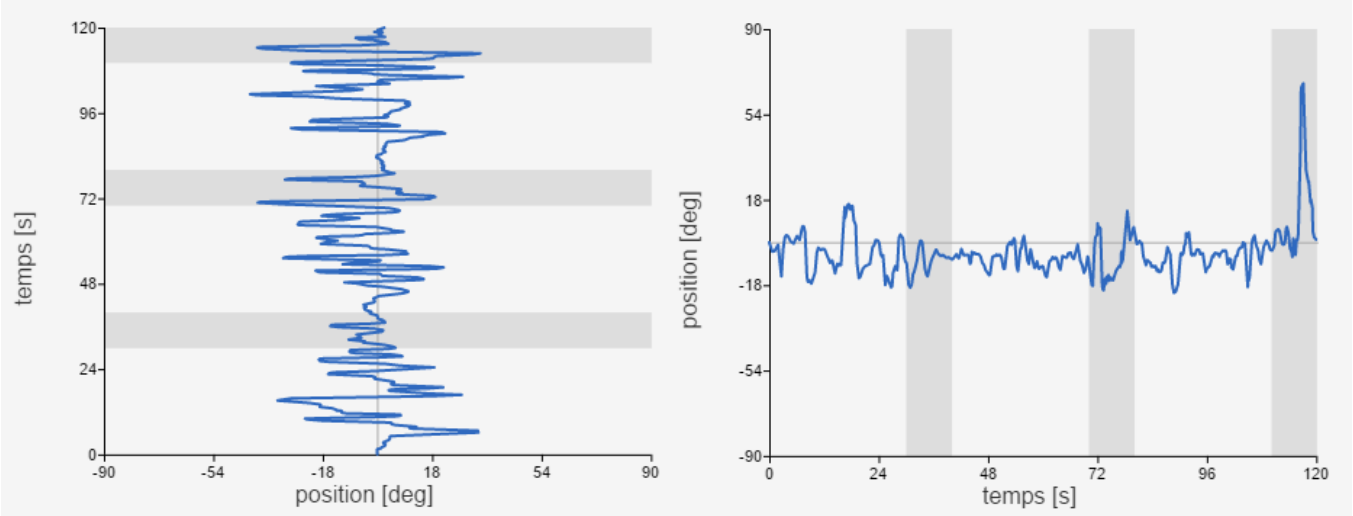


Current Visualization

Available:

- Average degree of body movement over x axis
- Average degree of body movement over y axis

Motivation

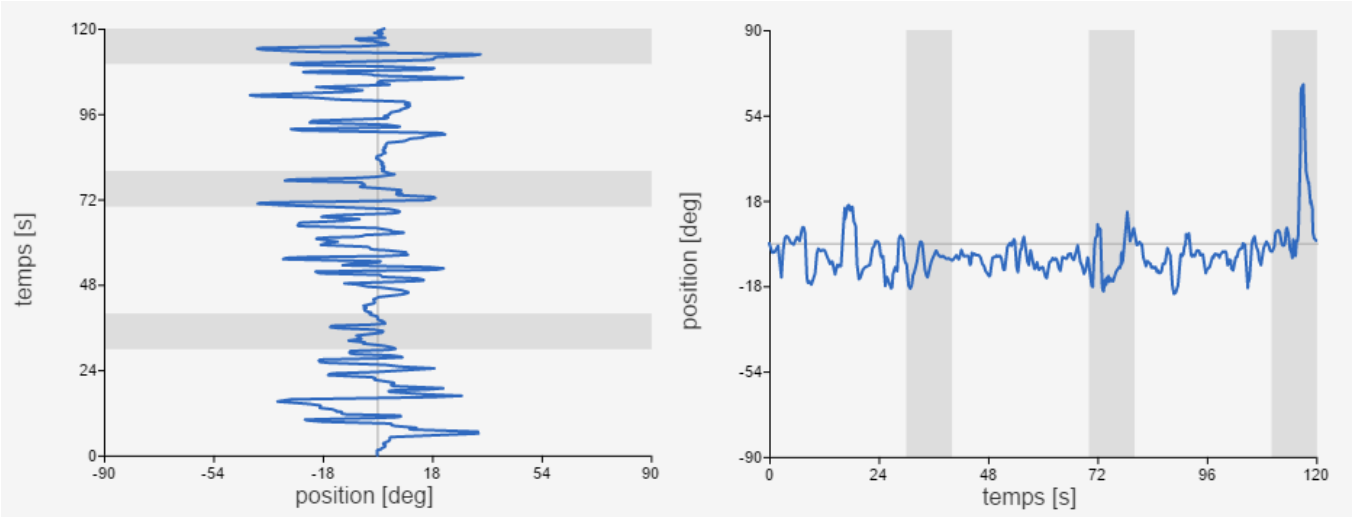


Current Visualization

Not available:

- How often the player move to right or left?

Motivation

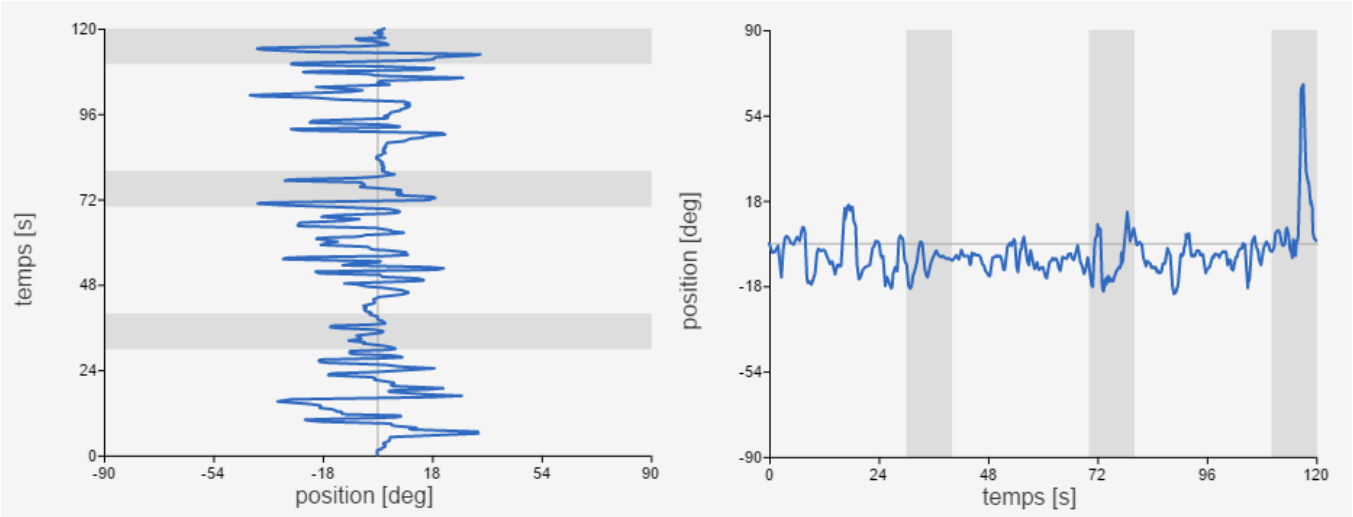


Current Visualization

Not available:

- How often the player move to right or left?
- To which type of events (collecting bonuses, killing enemies) the movement is related?

Motivation

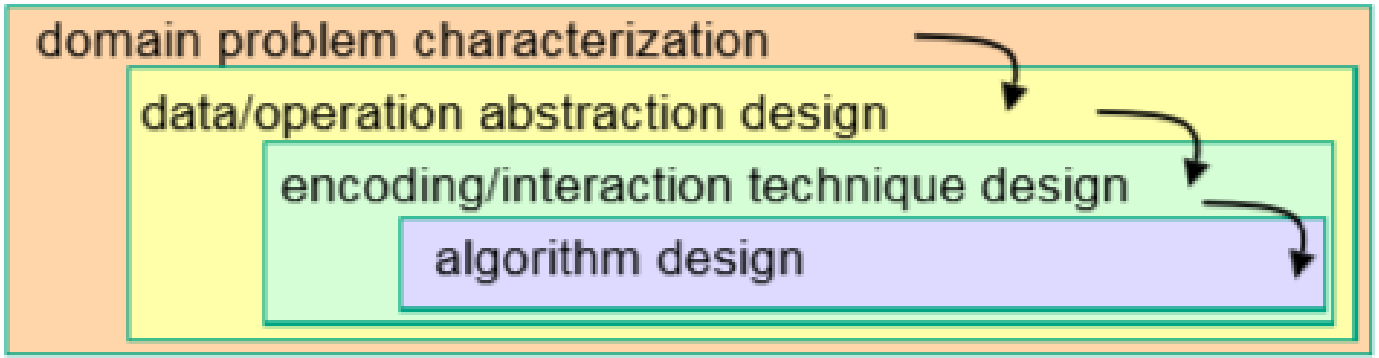


Current Visualization

Not available:

- How often the player move to right or left?
- To which type of events (collecting bonuses, killing enemies) the movement is related?
- Evolution of player's movement

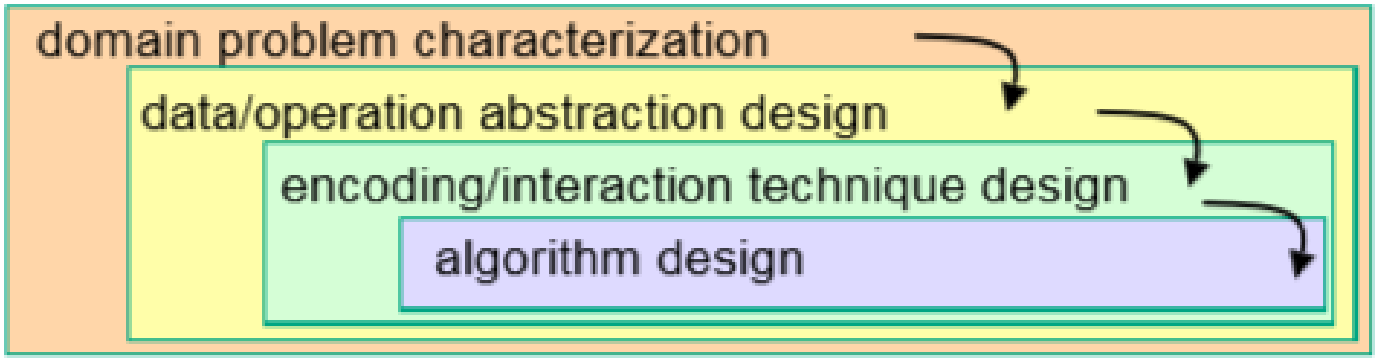
Methodology



Munzner's Visualization Design Model (Munzer, 2009)

- What kind of information needed by health professionals from the visualization (List of Tasks)

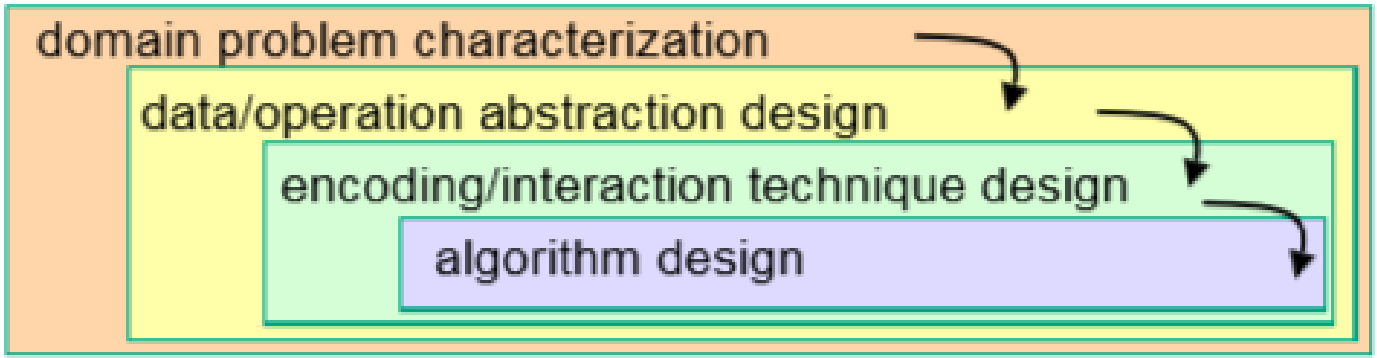
Methodology



Munzner's Visualization Design Model (Munzer, 2009)

- What kind of information needed by health professionals from the visualization (List of Tasks)
- Define data structure to support the tasks

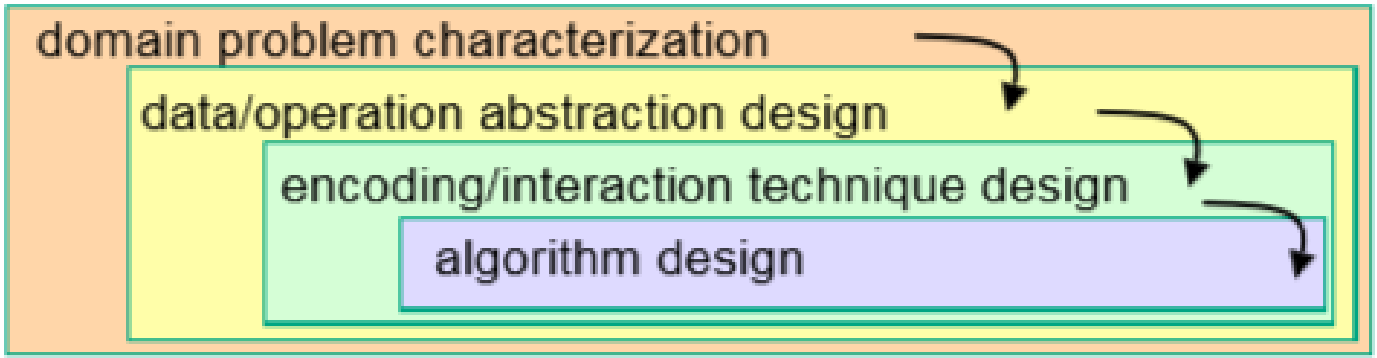
Methodology



Munzner's Visualization Design Model (Munzer, 2009)

- What kind of information needed by health professionals from the visualization (List of Tasks)
- Define data structure to support the tasks
- Define visualization and interaction technique

Methodology



Munzner's Visualization Design Model (Munzer, 2009)

- What kind of information needed by health professionals from the visualization (List of Tasks)
- Define data structure to support the tasks
- Define visualization and interaction technique
- Define algorithm to support the visualization

Hammer and Planks Game Dynamic

- Game difficulty can be changed by setting:
  - **Number of objects**
  - **Area where objects can appear**
  - **Activity duration and repetition**
  - **Movement Type**
  - **Movement Direction**

Hammer and Planks Game Dynamic

- Game difficulty can be changed by setting:
  - **Number of objects**
  - **Area where objects can appear**
  - **Activity duration and repetition**
  - **Movement Type**
  - **Movement Direction**
- Movement Type:
  - **BodyTilt**
  - **HandPoint**
  - **ShoulderCGE**

Hammer and Planks Game Dynamic

- Game difficulty can be changed by setting:
  - **Number of objects**
  - **Area where objects can appear**
  - **Activity duration and repetition**
  - **Movement Type**
  - **Movement Direction**
- Movement Type:
  - **BodyTilt**
  - **HandPoint**
  - **ShoulderCGE**
- Movement Direction:
  - **Horizontal**: moves boat horizontally
  - **Vertical**: changes boat speed

Hammer and Planks Game Dynamic

- Game difficulty can be changed by setting:
  - **Number of objects**
  - **Area where objects can appear**
  - **Activity duration and repetition**
  - **Movement Type**
  - **Movement Direction**
- Movement Type:
  - **BodyTilt**
  - **HandPoint**
  - **ShoulderCGE**
- Movement Direction:
  - **Horizontal**: moves boat horizontally
  - **Vertical**: changes boat speed

BodyTilt

Target User Question

(Q1) For a given session, to **which direction (right/left)** the player moved more?

Target User Question

(Q1) For a given session, to **which direction (right/left)** the player moved more?
(Q2) For a given session, how does the player perform based on the **number of objects collected, avoided, or killed** with respect to the **area of the movement**?

Target User Question

(Q1) For a given session, to **which direction (right/left)** the player moved more?
(Q2) For a given session, how does the player perform based on the **number of objects collected, avoided, or killed** with respect to the **area of the movement**?
(Q3) For a given session, how does the player perform based on the **number of objects collected, avoided, or killed** with respect to the **area of movement and the speed** in which the game is played?

Target User Question

(Q1) For a given session, to **which direction (right/left)** the player moved more?
(Q2) For a given session, how does the player perform based on the **number of objects collected, avoided, or killed** with respect to the **area of the movement**?
(Q3) For a given session, how does the player perform based on the **number of objects collected, avoided, or killed** with respect to the **area of movement and the speed** in which the game is played?
(Q4) For a given patient, has he/she **improved in the game overtime**?

Target User Question

(Q1) For a given session, to **which direction (right/left)** the player moved more?
(Q2) For a given session, how does the player perform based on the **number of objects collected, avoided, or killed** with respect to the **area of the movement**?
(Q3) For a given session, how does the player perform based on the **number of objects collected, avoided, or killed** with respect to the **area of movement and the speed** in which the game is played?
(Q4) For a given patient, has he/she **improved in the game overtime**?
(Q5) For a given patient, has he/she **improved in a certain area overtime**?

Game Events Structure

- **Event Category**
  - There are 8 types of events: catch, miss, dodge, etc.

Game Events Structure

- **Event Category**
  - There are 8 types of events: catch, miss, dodge, etc.
  - Categorized by its impact to player's boat into: **Positive**, **Neutral**, **Negative**

Game Events Structure

- **Event Category**
  - There are 8 types of events: catch, miss, dodge, etc.
  - Categorized by its impact to player's boat into: **Positive**, **Neutral**, **Negative**

Events	**Bonus**	**Obstacle**	**Enemy**
Positive	catch	-	kill,hit
Neutral	miss	dodge	miss
Negative	-	collision	hurt, collision

Game Events Structure

- **Event Category**
  - There are 8 types of events: catch, miss, dodge, etc.
  - Categorized by its impact to player's boat into: **Positive**, **Neutral**, **Negative**

Events	**Bonus**	**Obstacle**	**Enemy**
Positive	catch	-	kill,hit
Neutral	miss	dodge	miss
Negative	-	collision	hurt, collision

Game Events Structure

- **Event Category**
  - There are 8 types of events: catch, miss, dodge, etc.
  - Categorized by its impact to player's boat into: **Positive**, **Neutral**, **Negative**

Events	**Bonus**	**Obstacle**	**Enemy**
Positive	catch	-	kill,hit
Neutral	miss	dodge	miss
Negative	-	collision	hurt, collision

- **Screen Speed**: distance between apparition location and event location divided by duration between apparition time and event time

Game Events Structure

- **Event Category**
  - There are 8 types of events: catch, miss, dodge, etc.
  - Categorized by its impact to player's boat into: **Positive**, **Neutral**, **Negative**

Events	**Bonus**	**Obstacle**	**Enemy**
Positive	catch	-	kill,hit
Neutral	miss	dodge	miss
Negative	-	collision	hurt, collision

- **Screen Speed**: distance between apparition location and event location divided by duration between apparition time and event time

$$v_{scr} = \frac{\theta_{evt} - \theta_{apr}}{t_{evt} - t_{apr}}$$

Clustering - Distance Calculation

Clustering - Distance Calculation

$$s_1 = \begin{bmatrix} 10 & 20 & 6 \\ 20 & 5 & 18 \end{bmatrix} \quad s_2 = \begin{bmatrix} 20 & 40 & 10 \\ 40 & 10 & 30 \end{bmatrix} \quad s_3 = \begin{bmatrix} 10 & 20 & 5 \\ 16 & 4 & 12 \end{bmatrix}$$

Clustering - Distance Calculation

$$s_1 = \begin{bmatrix} 10 & 20 & 6 \\ 20 & 5 & 18 \end{bmatrix} s_2 = \begin{bmatrix} 20 & 40 & 10 \\ 40 & 10 & 30 \end{bmatrix} s_3 = \begin{bmatrix} 10 & 20 & 5 \\ 16 & 4 & 12 \end{bmatrix}$$

- Let S be a gameplay data set of n_{ses} sessions (Number of rows).

Clustering - Distance Calculation

$$s_1 = \begin{bmatrix} 10 & 20 & 6 \\ 20 & 5 & 18 \end{bmatrix} s_2 = \begin{bmatrix} 20 & 40 & 10 \\ 40 & 10 & 30 \end{bmatrix} s_3 = \begin{bmatrix} 10 & 20 & 5 \\ 16 & 4 & 12 \end{bmatrix}$$

- Let S be a gameplay data set of n_{ses} sessions (Number of rows).
- S is an ordered list of sections $s_i, 0 \leq i < n_{sec}$ (Number of Matrices)

Clustering - Distance Calculation

$$s_1 = \begin{bmatrix} 10 & 20 & 6 \\ 20 & 5 & 18 \end{bmatrix} s_2 = \begin{bmatrix} 20 & 40 & 10 \\ 40 & 10 & 30 \end{bmatrix} s_3 = \begin{bmatrix} 10 & 20 & 5 \\ 16 & 4 & 12 \end{bmatrix}$$

- Let S be a gameplay data set of n_{ses} sessions (Number of rows).
- S is an ordered list of sections $s_i, 0 \leq i < n_{sec}$ (Number of Matrices)
- A section s_i is a sequence of triplets.

Clustering - Distance Calculation

$$s_1 = \begin{bmatrix} 10 & 20 & 6 \\ 20 & 5 & 18 \end{bmatrix} s_2 = \begin{bmatrix} 20 & 40 & 10 \\ 40 & 10 & 30 \end{bmatrix} s_3 = \begin{bmatrix} 10 & 20 & 5 \\ 16 & 4 & 12 \end{bmatrix}$$

- Let S be a gameplay data set of n_{ses} sessions (Number of rows).
- S is an ordered list of sections $s_i, 0 \leq i < n_{sec}$ (Number of Matrices)
- A section s_i is a sequence of triplets.
- The triplet consists of the number of Positive, Neutral and Negative events.

Clustering - Distance Calculation

$$s_1 = \begin{bmatrix} 10 & 20 & 6 \\ 20 & 5 & 18 \end{bmatrix} s_2 = \begin{bmatrix} 20 & 40 & 10 \\ 40 & 10 & 30 \end{bmatrix} s_3 = \begin{bmatrix} 10 & 20 & 5 \\ 16 & 4 & 12 \end{bmatrix}$$

- Let S be a gameplay data set of n_{ses} sessions (Number of rows).
- S is an ordered list of sections $s_i, 0 \leq i < n_{sec}$ (Number of Matrices)
- A section s_i is a sequence of triplets.
- The triplet consists of the number of Positive, Neutral and Negative events.

Clustering - Distance Calculation

- Similarity between two sections is quantified with distance function (distance=0:similar; distance=1: different)

Clustering - Distance Calculation

- Similarity between two sections is quantified with distance function (distance=0:similar; distance=1: different)
- Two types of distance:

Clustering - Distance Calculation

- Similarity between two sections is quantified with distance function (distance=0:similar; distance=1: different)
- Two types of distance:
  - How different both sections in term of event types proportion within each section (s_2 is similar to s_3)

Clustering - Distance Calculation

- Similarity between two sections is quantified with distance function (distance=0:similar; distance=1: different)
- Two types of distance:
  - How different both sections in term of event types proportion within each section (s_2 is similar to s_3)
  - How different both sections in term of the evolution of each event type throughout the sessions (s_1 is similar to s_2)

Clustering - Distance Calculation

- Similarity between two sections is quantified with distance function (distance=0:similar; distance=1: different)
- Two types of distance:
  - How different both sections in term of event types proportion within each section (s_2 is similar to s_3)
  - How different both sections in term of the evolution of each event type throughout the sessions (s_1 is similar to s_2)

$$s_1 = \begin{bmatrix} 10 & 20 & 6 \\ 20 & 5 & 18 \end{bmatrix} \; s_2 = \begin{bmatrix} 20 & 40 & 10 \\ 40 & 10 & 30 \end{bmatrix} \; s_3 = \begin{bmatrix} 10 & 20 & 5 \\ 16 & 4 & 12 \end{bmatrix}$$

Clustering - Distance Calculation

- Similarity between two sections is quantified with distance function (distance=0:similar; distance=1: different)
- Two types of distance:
  - How different both sections in term of event types proportion within each section (s_2 is similar to s_3)
  - How different both sections in term of the evolution of each event type throughout the sessions (s_1 is similar to s_2)

$$s_1 = \begin{bmatrix} 10 & 20 & 6 \\ 20 & 5 & 18 \end{bmatrix} \; s_2 = \begin{bmatrix} 20 & 40 & 10 \\ 40 & 10 & 30 \end{bmatrix} \; s_3 = \begin{bmatrix} 10 & 20 & 5 \\ 16 & 4 & 12 \end{bmatrix}$$

Clustering - Distance Calculation

- Similarity between two sections is quantified with distance function (distance=0:similar; distance=1: different)
- Two types of distance:
  - How different both sections in term of event types proportion within each section (s_2 is similar to s_3)
  - How different both sections in term of the evolution of each event type throughout the sessions (s_1 is similar to s_2)

$$s_1 = \begin{bmatrix} 10 & 20 & 6 \\ 20 & 5 & 18 \end{bmatrix} s_2 = \begin{bmatrix} 20 & 40 & 10 \\ 40 & 10 & 30 \end{bmatrix} s_3 = \begin{bmatrix} 10 & 20 & 5 \\ 16 & 4 & 12 \end{bmatrix}$$

Clustering - Distance Calculation

- Similarity between two sections is quantified with distance function (distance=0:similar; distance=1: different)
- Two types of distance:
  - How different both sections in term of event types proportion within each section (s_2 is similar to s_3)
  - How different both sections in term of the evolution of each event type throughout the sessions (s_1 is similar to s_2)

$$s_1 = \begin{bmatrix} 10 & 20 & 6 \\ 20 & 5 & 18 \end{bmatrix} s_2 = \begin{bmatrix} 20 & 40 & 10 \\ 40 & 10 & 30 \end{bmatrix} s_3 = \begin{bmatrix} 10 & 20 & 5 \\ 16 & 4 & 12 \end{bmatrix}$$

- For each pair of consecutive sequences (s_1,s_2) of S, distance is weighted sum of two distance types, represented as $f(s_1, s_2)$ and $g(s_1, s_2)$.

$$d(s_1, s_2) = \alpha f(s_1, s_2) + (1 - \alpha) g(s_1, s_2)$$

Clustering - Distance Calculation

- $f(s_1, s_2)$ is a normalized euclidean distance of two triplets.

Clustering - Distance Calculation

- $f(s_1, s_2)$ is a normalized euclidean distance of two triplets.
- Distance is the average euclidean distance of each triplets pair.

$$f(s_1, s_2) = \frac{\sum_{i=0}^{i<|s_1|} NED(s_1[i], s_2[i])}{|s_1|}$$

Clustering - Distance Calculation

- $f(s_1, s_2)$ is a normalized euclidean distance of two triplets.
- Distance is the average euclidean distance of each triplets pair.

$$f(s_1, s_2) = \frac{\sum_{i=0}^{i<|s_1|} NED(s_1[i], s_2[i])}{|s_1|}$$

- Distance is normalized by dividing it with maximum distance $\sqrt{3}$.

$$NED(s_1[i], s_2[i]) = \frac{\sqrt{\sum_{j \in \{0,1,2\}} (s_1'[i][j] - s_2'[i][j])^2}}{\sqrt{3}}$$

Clustering - Distance Calculation

- $f(s_1, s_2)$ is a normalized euclidean distance of two triplets.
- Distance is the average euclidean distance of each triplets pair.

$$f(s_1, s_2) = \frac{\sum_{i=0}^{i<|s_1|} NED(s_1[i], s_2[i])}{|s_1|}$$

- Distance is normalized by dividing it with maximum distance $\sqrt{3}$.

$$NED(s_1[i], s_2[i]) = \frac{\sqrt{\sum_{j \in \{0,1,2\}} (s'_1[i][j] - s'_2[i][j])^2}}{\sqrt{3}}$$

- s'_1 and s'_2 is normalized value of s_1 and s_2 (divided by max value of each session in each section)

Clustering - Distance Calculation

$$s_1 = \begin{bmatrix} 10 & 20 & 6 \\ 20 & 5 & 18 \end{bmatrix} s_2 = \begin{bmatrix} \mathbf{20} & \mathbf{40} & \mathbf{10} \\ 40 & 10 & 30 \end{bmatrix} s_3 = \begin{bmatrix} \mathbf{10} & \mathbf{20} & \mathbf{5} \\ 16 & 4 & 12 \end{bmatrix}$$

Clustering - Distance Calculation

$$s_1 = \begin{bmatrix} 10 & 20 & 6 \\ 20 & 5 & 18 \end{bmatrix} s_2 = \begin{bmatrix} \mathbf{20} & \mathbf{40} & \mathbf{10} \\ 40 & 10 & 30 \end{bmatrix} s_3 = \begin{bmatrix} \mathbf{10} & \mathbf{20} & \mathbf{5} \\ 16 & 4 & 12 \end{bmatrix}$$

Clustering - Distance Calculation

$$s_1 = \begin{bmatrix} 10 & 20 & 6 \\ 20 & 5 & 18 \end{bmatrix} s_2 = \begin{bmatrix} 20 & 40 & 10 \\ \mathbf{40} & \mathbf{10} & \mathbf{30} \end{bmatrix} s_3 = \begin{bmatrix} 10 & 20 & 5 \\ \mathbf{16} & \mathbf{4} & \mathbf{12} \end{bmatrix}$$

Clustering - Distance Calculation

$$s_1 = \begin{bmatrix} 10 & 20 & 6 \\ 20 & 5 & 18 \end{bmatrix} \; s_2 = \begin{bmatrix} \mathbf{20} & \mathbf{40} & \mathbf{10} \\ \mathbf{40} & \mathbf{10} & \mathbf{30} \end{bmatrix} \; s_3 = \begin{bmatrix} \mathbf{10} & \mathbf{20} & \mathbf{5} \\ \mathbf{16} & \mathbf{4} & \mathbf{12} \end{bmatrix}$$

$$s_1' = \begin{bmatrix} \frac{1}{2} & 1 & \frac{6}{20} \\ 1 & \frac{1}{4} & \frac{18}{20} \end{bmatrix} \; s_2' = \begin{bmatrix} \mathbf{\frac{1}{2}} & \mathbf{1} & \mathbf{\frac{1}{4}} \\ \mathbf{1} & \mathbf{\frac{1}{4}} & \mathbf{\frac{3}{4}} \end{bmatrix} \; s_3' = \begin{bmatrix} \mathbf{\frac{1}{2}} & \mathbf{1} & \mathbf{\frac{1}{4}} \\ \mathbf{1} & \mathbf{\frac{1}{4}} & \mathbf{\frac{3}{4}} \end{bmatrix}$$

Clustering - Distance Calculation

$$s_1 = \begin{bmatrix} 10 & 20 & 6 \\ 20 & 5 & 18 \end{bmatrix} \; s_2 = \begin{bmatrix} \mathbf{20} & \mathbf{40} & \mathbf{10} \\ \mathbf{40} & \mathbf{10} & \mathbf{30} \end{bmatrix} \; s_3 = \begin{bmatrix} \mathbf{10} & \mathbf{20} & \mathbf{5} \\ \mathbf{16} & \mathbf{4} & \mathbf{12} \end{bmatrix}$$

$$s_1' = \begin{bmatrix} \frac{1}{2} & 1 & \frac{6}{20} \\ 1 & \frac{1}{4} & \frac{18}{20} \end{bmatrix} \; s_2' = \begin{bmatrix} \mathbf{\frac{1}{2}} & \mathbf{1} & \mathbf{\frac{1}{4}} \\ \mathbf{1} & \mathbf{\frac{1}{4}} & \mathbf{\frac{3}{4}} \end{bmatrix} \; s_3' = \begin{bmatrix} \mathbf{\frac{1}{2}} & \mathbf{1} & \mathbf{\frac{1}{4}} \\ \mathbf{1} & \mathbf{\frac{1}{4}} & \mathbf{\frac{3}{4}} \end{bmatrix}$$

Clustering - Distance Calculation

- $g(s_1, s_2)$ is based on a normalized euclidean distance between the same event type j from different section.

Clustering - Distance Calculation

- $g(s_1, s_2)$ is based on a normalized euclidean distance between the same event type j from different section.
- The overall distance of both sections is the average euclidean distance of each event type pair.

$$g(s_1, s_2) = \frac{g_0(s_1, s_2) + g_1(s_1, s_2) + g_2(s_1, s_2)}{3}$$

Clustering - Distance Calculation

- $g(s_1, s_2)$ is based on a normalized euclidean distance between the same event type j from different section.
- The overall distance of both sections is the average euclidean distance of each event type pair.

$$g(s_1, s_2) = \frac{g_0(s_1, s_2) + g_1(s_1, s_2) + g_2(s_1, s_2)}{3}$$

- For each event type distance, there are three different distance value defined:

Clustering - Distance Calculation

- $g(s_1, s_2)$ is based on a normalized euclidean distance between the same event type j from different section.
- The overall distance of both sections is the average euclidean distance of each event type pair.

$$g(s_1, s_2) = \frac{g_0(s_1, s_2) + g_1(s_1, s_2) + g_2(s_1, s_2)}{3}$$

- For each event type distance, there are three different distance value defined:
  - if there are no events in both event type pairs, the distance is 0

Clustering - Distance Calculation

- $g(s_1, s_2)$ is based on a normalized euclidean distance between the same event type j from different section.
- The overall distance of both sections is the average euclidean distance of each event type pair.

$$g(s_1, s_2) = \frac{g_0(s_1, s_2) + g_1(s_1, s_2) + g_2(s_1, s_2)}{3}$$

- For each event type distance, there are three different distance value defined:
  - if there are no events in both event type pairs, the distance is 0
  - if there is at least one event in one section and there are no events in the other section, the distance is 1

Clustering - Distance Calculation

- $g(s_1, s_2)$ is based on a normalized euclidean distance between the same event type j from different section.
- The overall distance of both sections is the average euclidean distance of each event type pair.

$$g(s_1, s_2) = \frac{g_0(s_1, s_2) + g_1(s_1, s_2) + g_2(s_1, s_2)}{3}$$

- For each event type distance, there are three different distance value defined:
  - if there are no events in both event type pairs, the distance is 0
  - if there is at least one event in one section and there are no events in the other section, the distance is 1
  - if both event type pairs has any events then the euclidean distance is calculated

Clustering - Distance Calculation

for $j \in \{0, 1, 2\}$,

$$g_j(s_1, s_2) = \begin{cases} \end{cases}$$

Clustering - Distance Calculation

for $j \in \{0, 1, 2\}$,

$$g_j(s_1, s_2) = \begin{cases} 0 & \text{if } s_k[i][j] = 0 \text{ for each } 0 \leqslant j < |s_k| \text{ and } k \in \{1, 2\} \\ & \\ & \end{cases}$$

Clustering - Distance Calculation

for $j \in \{0, 1, 2\}$,

$$g_j(s_1, s_2) = \begin{cases} 0 & \text{if } s_k[i][j] = 0 \text{ for each } 0 \leqslant j < |s_k| \text{ and } k \in \{1, 2\} \\ 1 & \text{if } s_k[i][j] = 0 \text{ for each } 0 \leqslant j < |s_k| \text{ and } \exists s_p[i][j] \neq 0, \\ & k, p \in \{1, 2\},\ k \neq p \end{cases}$$

Clustering - Distance Calculation

for $j \in \{0, 1, 2\}$,

$$g_j(s_1, s_2) = \begin{cases} 0 & \text{if } s_k[i][j] = 0 \text{ for each } 0 \leqslant j < |s_k| \text{ and } k \in \{1, 2\} \\ 1 & \text{if } s_k[i][j] = 0 \text{ for each } 0 \leqslant j < |s_k| \text{ and } \exists s_p[i][j] \neq 0, \\ & k, p \in \{1, 2\},\ k \neq p \\ \dfrac{\sqrt{\sum_{i=0}^{|s_1|} (s_1''[i][j] - s_2''[i][j])^2}}{\sqrt{|s_1|}} & \text{otherwise} \end{cases}$$

Clustering - Distance Calculation

for $j \in \{0, 1, 2\}$,

$$g_j(s_1, s_2) = \begin{cases} 0 & \text{if } s_k[i][j] = 0 \text{ for each } 0 \leqslant j < |s_k| \text{ and } k \in \{1, 2\} \\ 1 & \text{if } s_k[i][j] = 0 \text{ for each } 0 \leqslant j < |s_k| \text{ and } \exists s_p[i][j] \neq 0, \\ & k, p \in \{1, 2\},\ k \neq p \\ \dfrac{\sqrt{\sum_{i=0}^{|s_1|} (s_1''[i][j] - s_2''[i][j])^2}}{\sqrt{|s_1|}} & \text{otherwise} \end{cases}$$

- s_1'' and s_2'' represent normalized value of s_1 and s_2 (divided by max value of each event type in each section)

Distance Calculation

$$s_1 = \begin{bmatrix} \mathbf{10} & 20 & 6 \\ \mathbf{20} & 5 & 18 \end{bmatrix} s_2 = \begin{bmatrix} \mathbf{20} & 40 & 10 \\ \mathbf{40} & 10 & 30 \end{bmatrix} s_3 = \begin{bmatrix} 10 & 20 & 5 \\ 16 & 4 & 12 \end{bmatrix}$$

Distance Calculation

$$s_1 = \begin{bmatrix} \mathbf{10} & 20 & 6 \\ \mathbf{20} & 5 & 18 \end{bmatrix} s_2 = \begin{bmatrix} \mathbf{20} & 40 & 10 \\ \mathbf{40} & 10 & 30 \end{bmatrix} s_3 = \begin{bmatrix} 10 & 20 & 5 \\ 16 & 4 & 12 \end{bmatrix}$$

Distance Calculation

$$s_1 = \begin{bmatrix} 10 & \mathbf{20} & 6 \\ 20 & \mathbf{5} & 18 \end{bmatrix} s_2 = \begin{bmatrix} 20 & \mathbf{40} & 10 \\ 40 & \mathbf{10} & 30 \end{bmatrix} s_3 = \begin{bmatrix} 10 & 20 & 5 \\ 16 & 4 & 12 \end{bmatrix}$$

Distance Calculation

$$s_1 = \begin{bmatrix} 10 & \mathbf{20} & 6 \\ 20 & \mathbf{5} & 18 \end{bmatrix} \quad s_2 = \begin{bmatrix} 20 & \mathbf{40} & 10 \\ 40 & \mathbf{10} & 30 \end{bmatrix} \quad s_3 = \begin{bmatrix} 10 & 20 & 5 \\ 16 & 4 & 12 \end{bmatrix}$$

Distance Calculation

$$s_1 = \begin{bmatrix} 10 & 20 & \mathbf{6} \\ 20 & 5 & \mathbf{18} \end{bmatrix} \quad s_2 = \begin{bmatrix} 20 & 40 & \mathbf{10} \\ 40 & 10 & \mathbf{30} \end{bmatrix} \quad s_3 = \begin{bmatrix} 10 & 20 & 5 \\ 16 & 4 & 12 \end{bmatrix}$$

Distance Calculation

$$s_1 = \begin{bmatrix} 10 & 20 & \mathbf{6} \\ 20 & 5 & \mathbf{18} \end{bmatrix} \; s_2 = \begin{bmatrix} 20 & 40 & \mathbf{10} \\ 40 & 10 & \mathbf{30} \end{bmatrix} \; s_3 = \begin{bmatrix} 10 & 20 & 5 \\ 16 & 4 & 12 \end{bmatrix}$$

Distance Calculation

$$s_1 = \begin{bmatrix} \mathbf{10} & \mathbf{20} & \mathbf{6} \\ \mathbf{20} & \mathbf{5} & \mathbf{18} \end{bmatrix} s_2 = \begin{bmatrix} \mathbf{20} & \mathbf{40} & \mathbf{10} \\ \mathbf{40} & \mathbf{10} & \mathbf{30} \end{bmatrix} s_3 = \begin{bmatrix} 10 & 20 & 5 \\ 16 & 4 & 12 \end{bmatrix}$$

$$s_1'' = \begin{bmatrix} \mathbf{\frac{1}{2}} & \mathbf{1} & \mathbf{\frac{1}{3}} \\ \mathbf{1} & \mathbf{\frac{1}{4}} & \mathbf{1} \end{bmatrix} s_2'' = \begin{bmatrix} \mathbf{\frac{1}{2}} & \mathbf{1} & \mathbf{\frac{1}{3}} \\ \mathbf{1} & \mathbf{\frac{1}{4}} & \mathbf{1} \end{bmatrix} s_3'' = \begin{bmatrix} \frac{5}{8} & 1 & \frac{5}{12} \\ 1 & \frac{1}{5} & 1 \end{bmatrix}$$

Distance Calculation

$$s_1 = \begin{bmatrix} \mathbf{10} & \mathbf{20} & \mathbf{6} \\ \mathbf{20} & \mathbf{5} & \mathbf{18} \end{bmatrix} s_2 = \begin{bmatrix} \mathbf{20} & \mathbf{40} & \mathbf{10} \\ \mathbf{40} & \mathbf{10} & \mathbf{30} \end{bmatrix} s_3 = \begin{bmatrix} 10 & 20 & 5 \\ 16 & 4 & 12 \end{bmatrix}$$

$$s_1'' = \begin{bmatrix} \mathbf{\frac{1}{2}} & \mathbf{1} & \mathbf{\frac{1}{3}} \\ \mathbf{1} & \mathbf{\frac{1}{4}} & \mathbf{1} \end{bmatrix} s_2'' = \begin{bmatrix} \mathbf{\frac{1}{2}} & \mathbf{1} & \mathbf{\frac{1}{3}} \\ \mathbf{1} & \mathbf{\frac{1}{4}} & \mathbf{1} \end{bmatrix} s_3'' = \begin{bmatrix} \frac{5}{8} & 1 & \frac{5}{12} \\ 1 & \frac{1}{5} & 1 \end{bmatrix}$$

Distance Calculation

$$s_1 = \begin{bmatrix} 10 & 20 & 6 \\ 20 & 5 & 18 \end{bmatrix} s_2 = \begin{bmatrix} 20 & 40 & 10 \\ 40 & 10 & 30 \end{bmatrix} s_3 = \begin{bmatrix} 10 & 20 & 5 \\ 16 & 4 & 12 \end{bmatrix}$$

$$s_1'' = \begin{bmatrix} \frac{1}{2} & 1 & \frac{1}{3} \\ 1 & \frac{1}{4} & 1 \end{bmatrix} s_2'' = \begin{bmatrix} \frac{1}{2} & 1 & \frac{1}{3} \\ 1 & \frac{1}{4} & 1 \end{bmatrix} s_3'' = \begin{bmatrix} \frac{5}{8} & 1 & \frac{5}{12} \\ 1 & \frac{1}{5} & 1 \end{bmatrix}$$

Clustering Algorithm

1. Divide data set into sections the size of x-axis unit

Clustering Algorithm

1. Divide data set into sections the size of x-axis unit
2. Calculate distance between two consecutive sections

Clustering Algorithm

1. Divide data set into sections the size of x-axis unit
2. Calculate distance between two consecutive sections
3. Cluster sections with distance below threshold. This results in a new set of sections

Clustering Algorithm

1. Divide data set into sections the size of x-axis unit
2. Calculate distance between two consecutive sections
3. Cluster sections with distance below threshold. This results in a new set of sections
4. Recalculate distance and cluster

Clustering Algorithm

1. Divide data set into sections the size of x-axis unit
2. Calculate distance between two consecutive sections
3. Cluster sections with distance below threshold. This results in a new set of sections
4. Recalculate distance and cluster
5. Repeat the process until there is no sections with distance below the threshold

Demo

Conclusion

- The proposed visualizations are able to provide information on movement direction, movement and its related events, movement evolution

Conclusion

- The proposed visualizations are able to provide information on movement direction, movement and its related events, movement evolution
- Case study shows the effectiveness of the interface in answering the questions asked by healthcare professionals

Conclusion

- The proposed visualizations are able to provide information on movement direction, movement and its related events, movement evolution
- Case study shows the effectiveness of the interface in answering the questions asked by healthcare professionals
- Future Works:

Conclusion

- The proposed visualizations are able to provide information on movement direction, movement and its related events, movement evolution
- Case study shows the effectiveness of the interface in answering the questions asked by healthcare professionals
- Future Works:
  - Improve clustering using different distance measure

Conclusion

- The proposed visualizations are able to provide information on movement direction, movement and its related events, movement evolution
- Case study shows the effectiveness of the interface in answering the questions asked by healthcare professionals
- Future Works:
  - Improve clustering using different distance measure
  - Study on pathology and its movement to propose game setting based on pathology and its severity

Conclusion

- The proposed visualizations are able to provide information on movement direction, movement and its related events, movement evolution
- Case study shows the effectiveness of the interface in answering the questions asked by healthcare professionals
- Future Works:
  - Improve clustering using different distance measure
  - Study on pathology and its movement to propose game setting based on pathology and its severity
  - Visualize log data related to skeleton movement

Question


thank you!

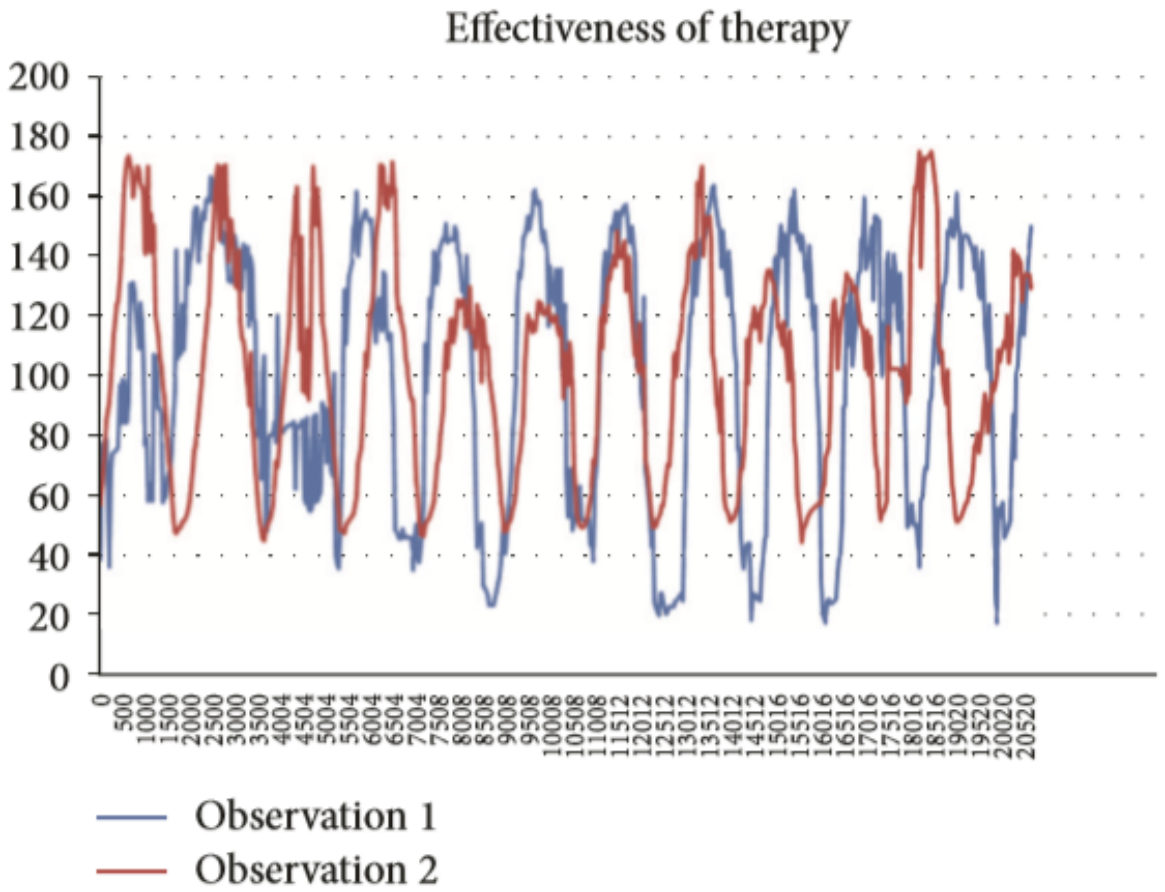
Visualization Requirements

Tasks related to a session of a particular player:
(T1.1) visualize and be able to compare the number of events within the same or among different event type at a given x area (Q1)(Q2).
(T1.2) visualize and be able to compare the number of events and its screen speed of the same or among different event type at a given x area (Q1)(Q2)(Q3).
(T1.3) select and visualize the number of events for a certain object at a given x area (Q1)(Q2).
(T1.4) select and visualize the number of events and its screen speed for a certain object at a given x area (Q1)(Q2)(Q3).

Visualization Requirements

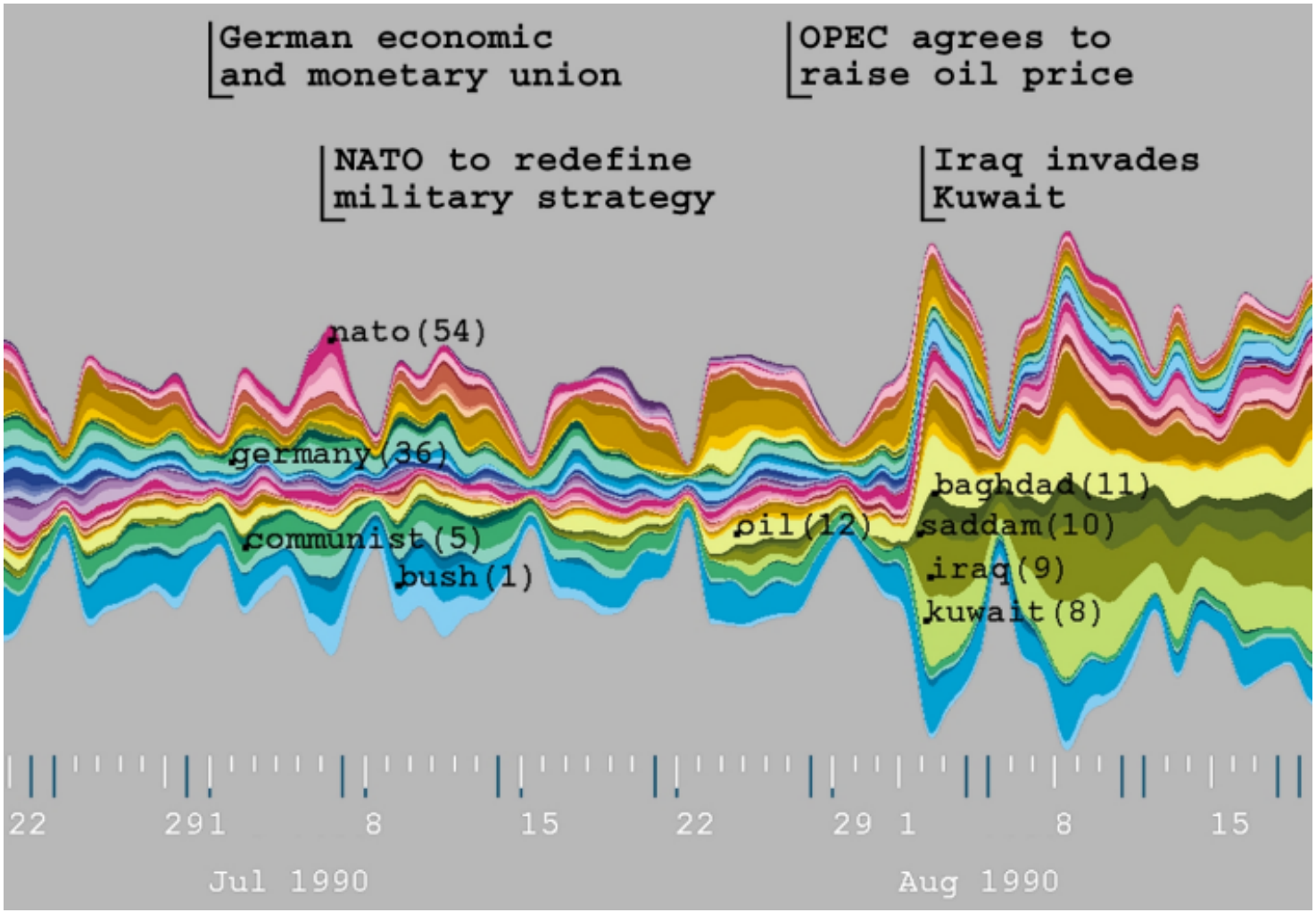
Tasks related to the summary of all sessions of a player:
(T2.1) visualize, navigate and be able to compare the evolution of number of events throughout all sessions within a certain x area.(Q4)(Q5).
(T2.2) select and visualize the number of events of a certain event type in a certain x area throughout all sessions (Q4)(Q5).
(T2.3) visualize, navigate and be able to compare the distribution of a certain number of events over x area among all sessions(Q4)(Q5).
(T2.4) select and visualize the distribution of certain number of events over x area for a certain event type throughout all sessions (Q4)(Q5).
(T2.5) extract and visualize similar pattern of number of events evolution throughout all sessions over a certain x area (Q4)(Q5).

Visualization of Serious Game Result



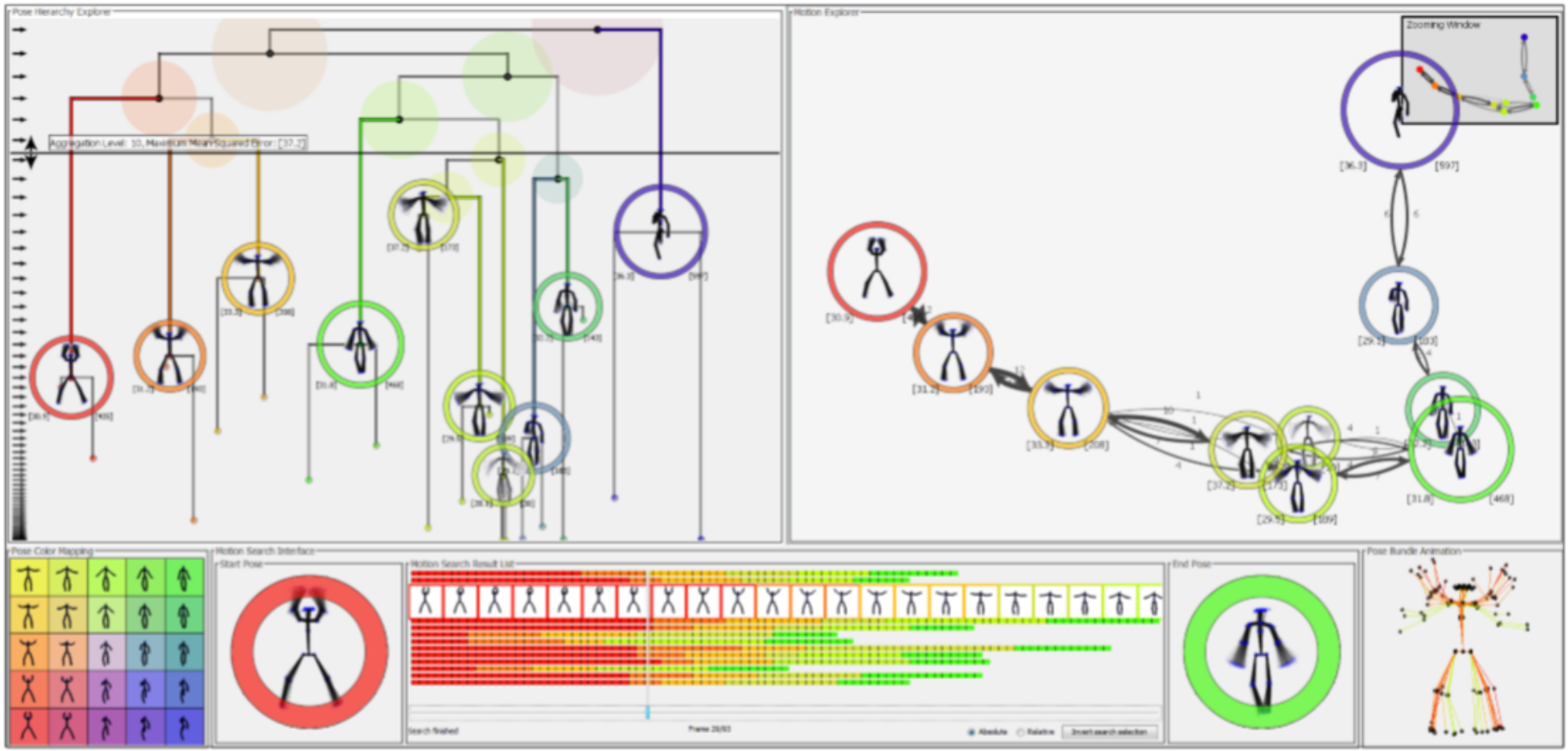
Line Chart depicting degree of forearm movement over time

Visualization of Time Series Data



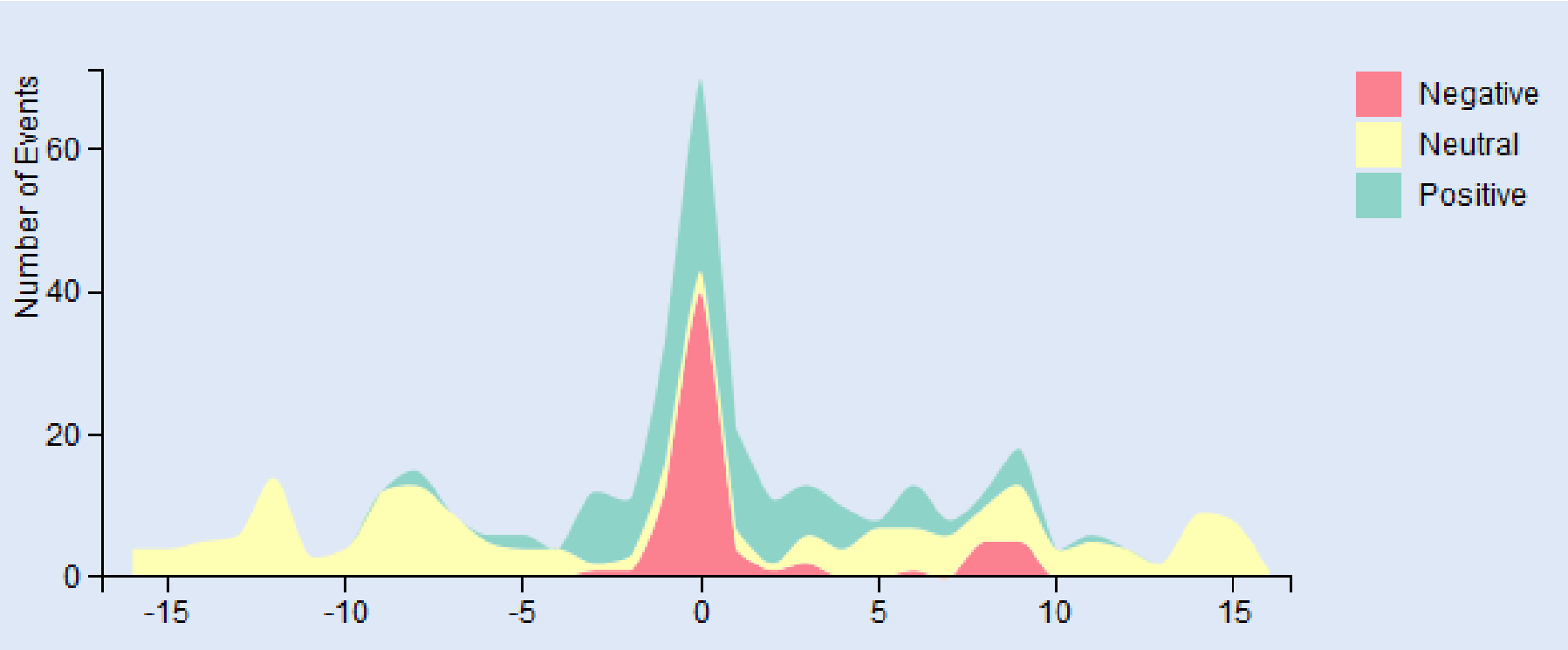
Theme River

Visualization of Movement Data



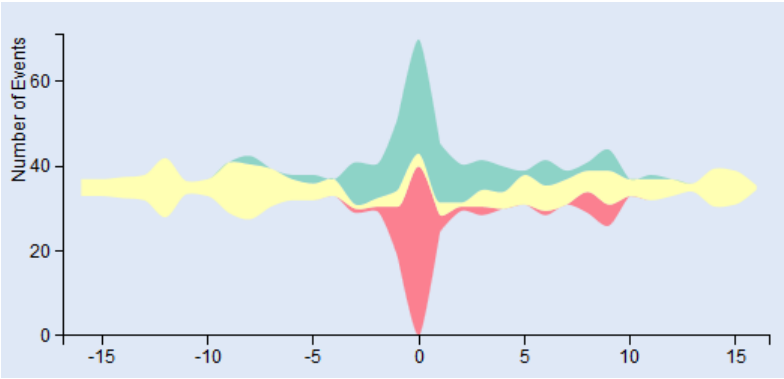
Motion Explorer

Session Visualization - Stacked Graph (T1.1)

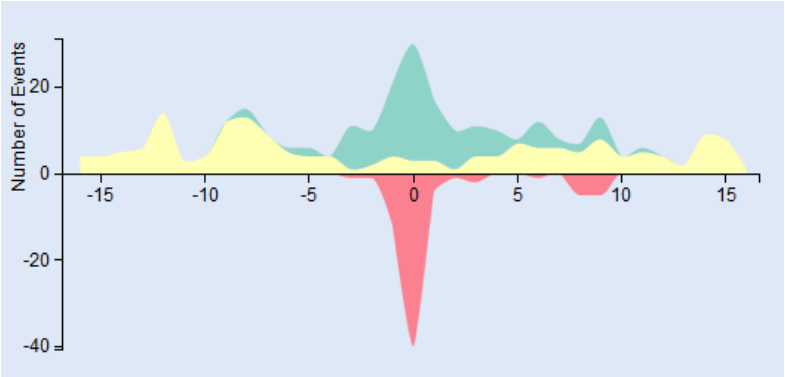


Stacked Graph depicting number of events over x axis

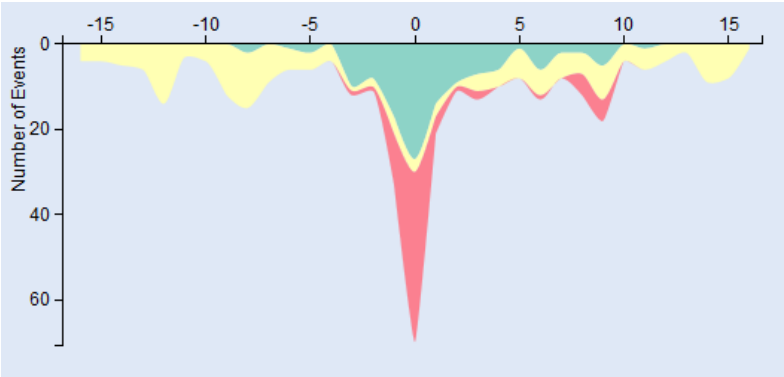
Session Visualization - Stacked Graph Layout



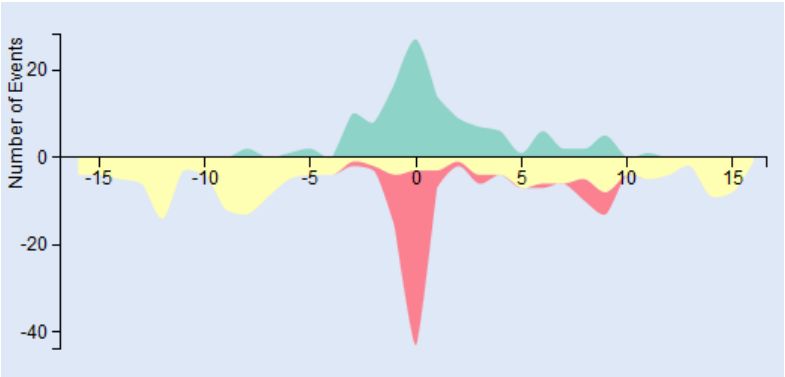
Silhouette



Neutral-Negative

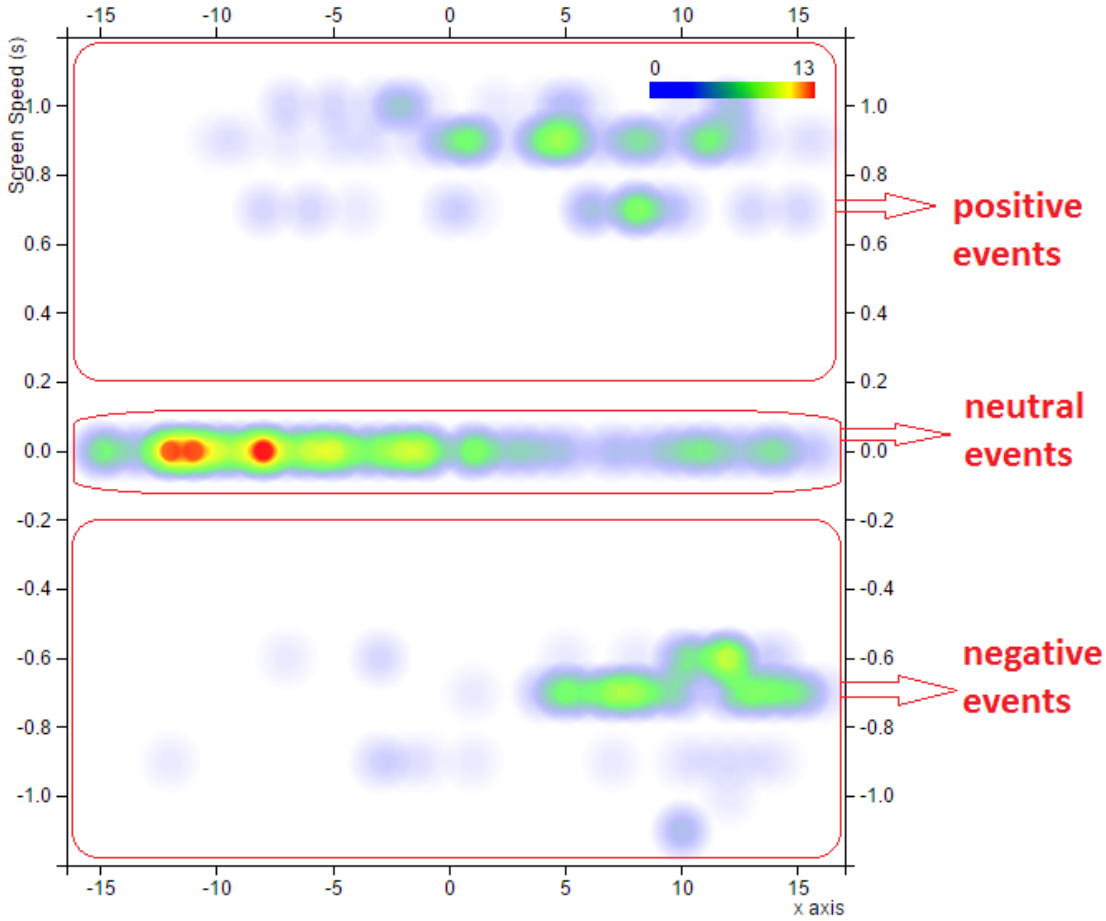


Positive



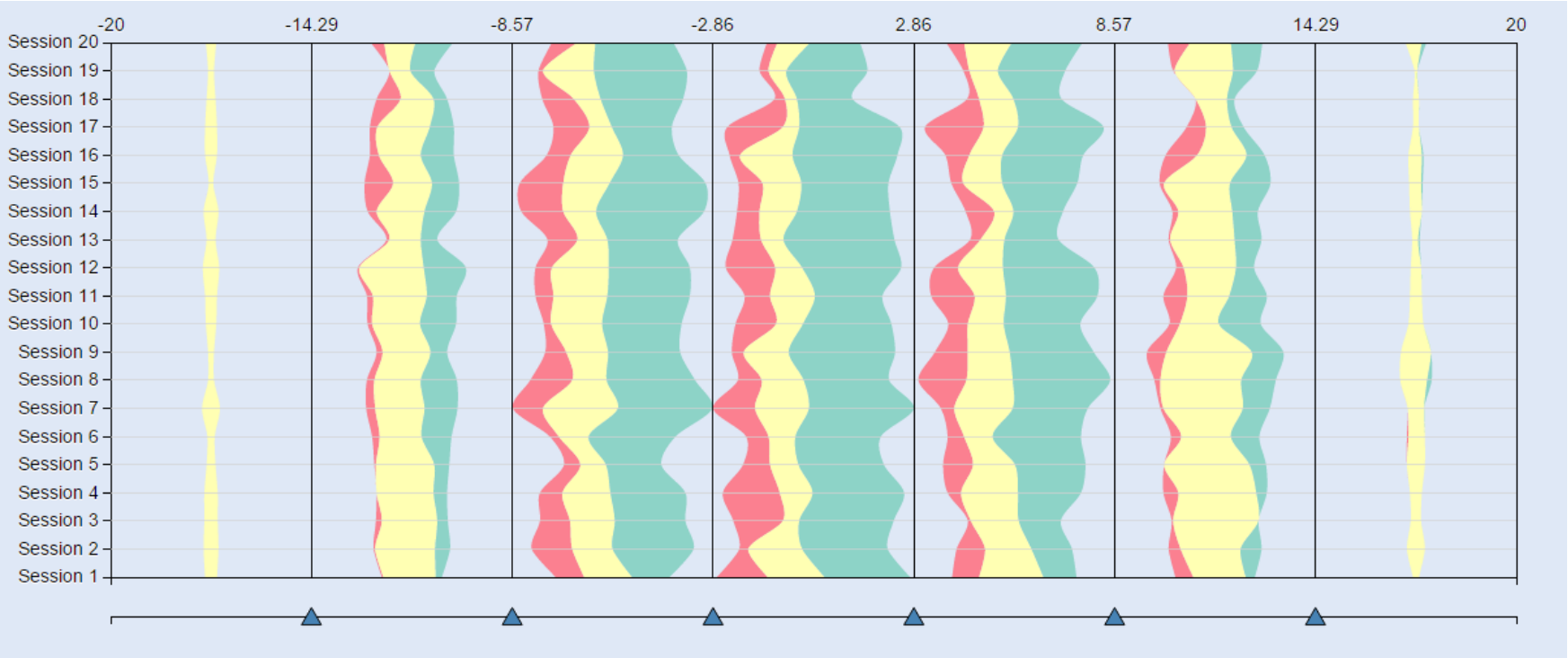
Positive-Neutral

Session Visualization - Heatmap (T1.2)



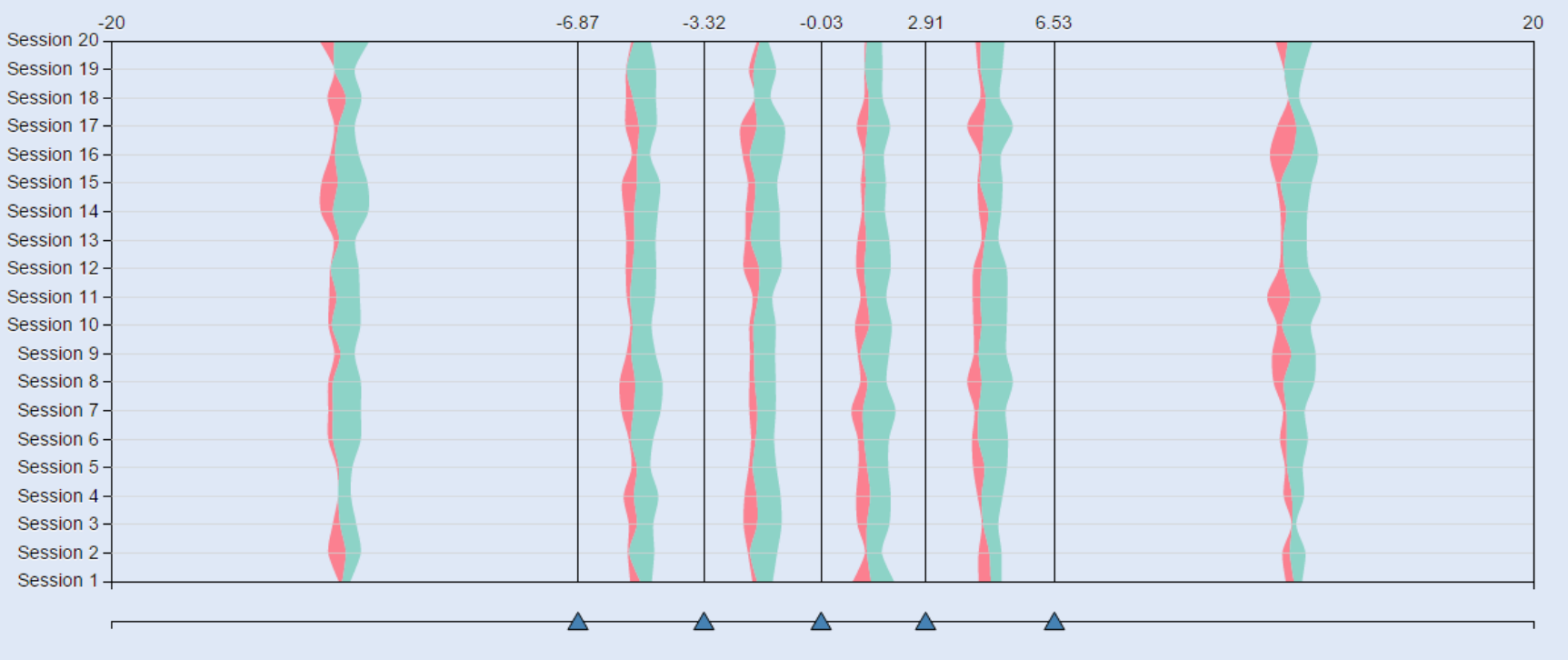
Heatmap depicting number of events and screen speed over x axis

Summary Visualization by range of x-area (T2.1)



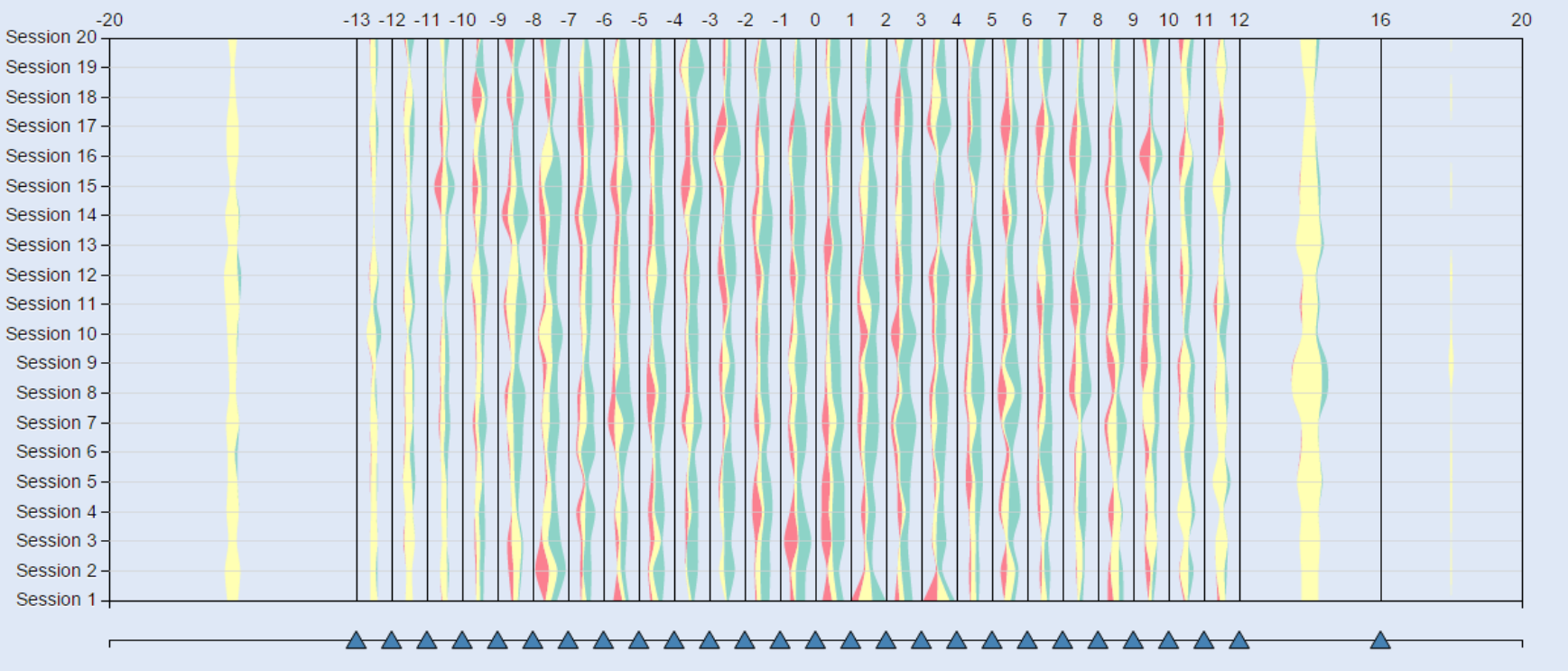
Summary Visualization by range of x-area: each section has the same x-range

Summary Visualization by number of events (T2.3)



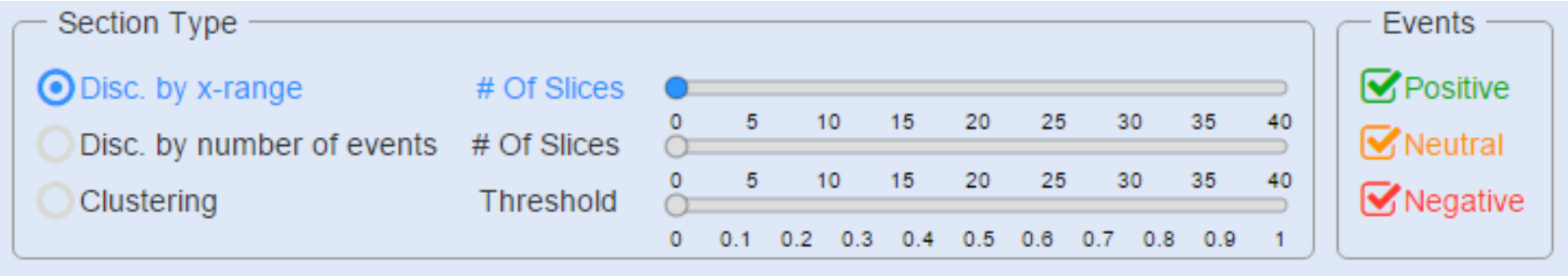
Summary Visualization by number of events: each section has the same number of positive and negative events

Summary Visualization by clustering (T2.5)



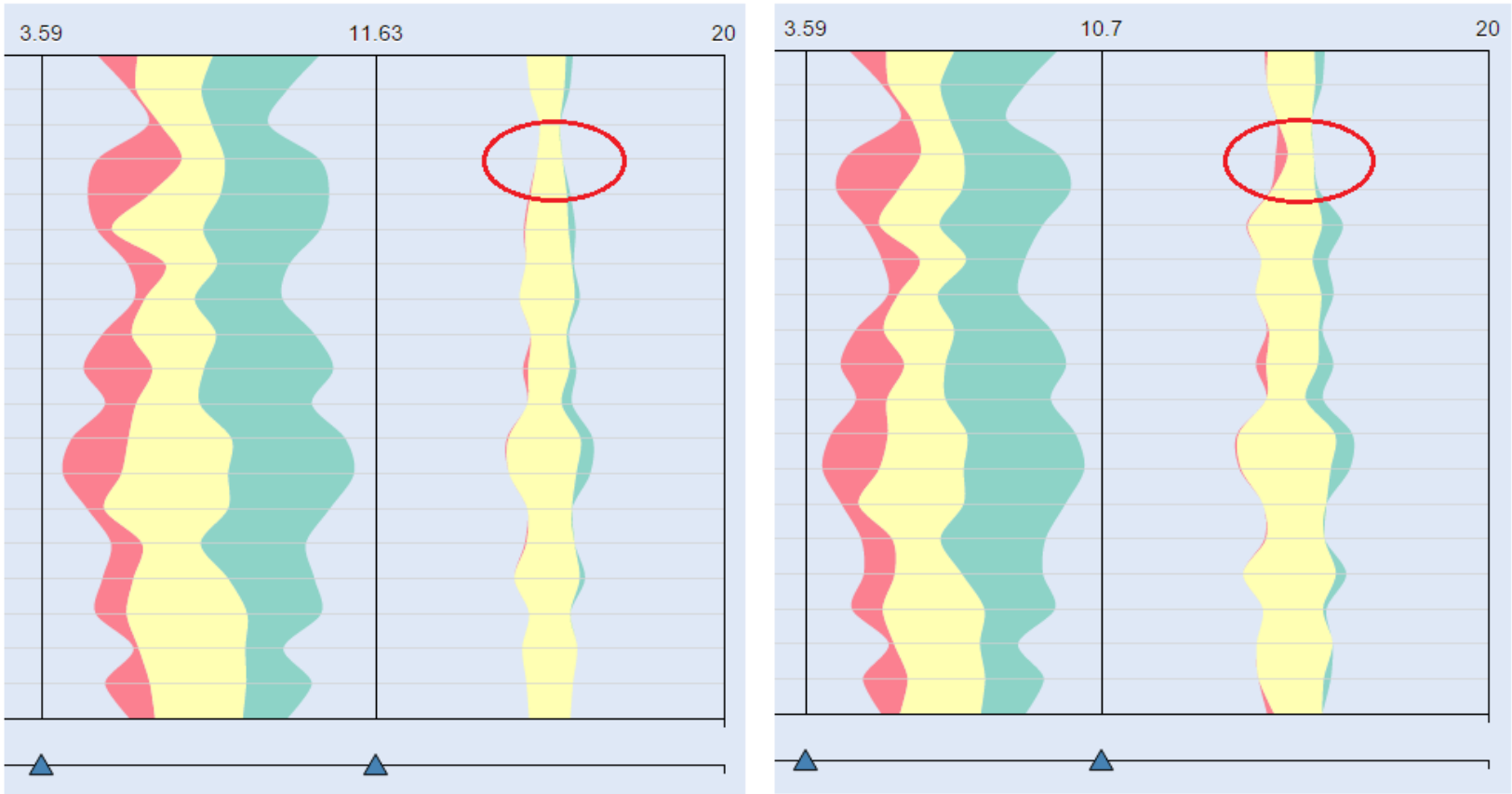
Summary Visualization by clustering: each section has the similar movement pattern

Summary Visualization - Interaction Technique(T2.2,T2.4)



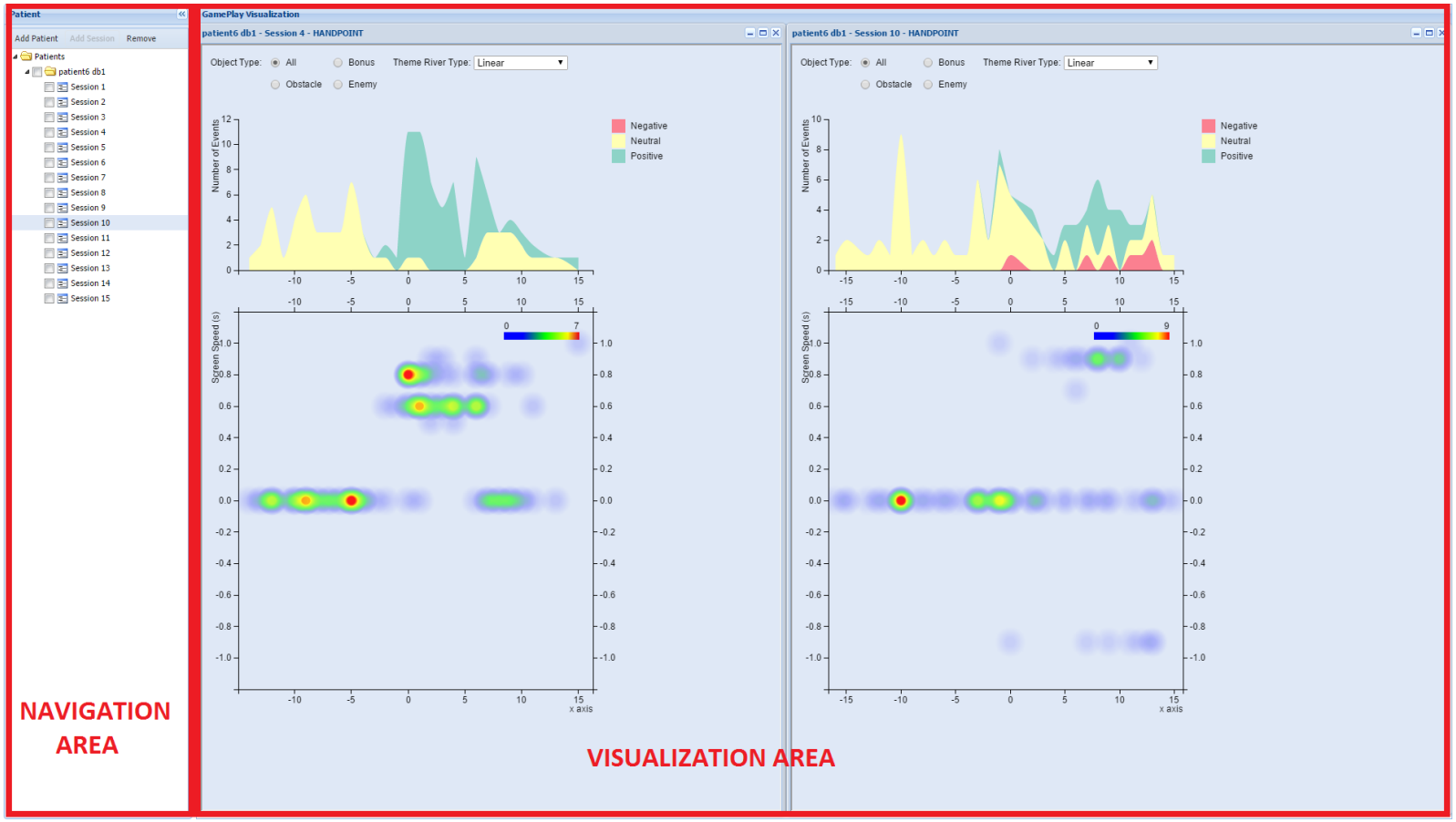
Interaction bar allows user to choose which event type to show and change input using sliders

Summary Visualization - Interaction Technique



By dragging section line or triangle symbol, user can highlight movement pattern

General Interface



Interface is divided into two areas: navigation area and visualization area

Case Study and Demo

- Data set is acquired from NaturalPad
- Data set is of game played by patient with pathology(type of pathology unknown)
- Case study using HandPoint exercise, consists of 6 sessions