7. Motion Capture and Character Animation

Game Engineering & XR Technologies
Prof. HyeongYeop Kang
siamiz@khu.ac.kr
IIIXR LAB

Motion Capture

Motion Capture[MC]?

- Motion capture (MoCap) is the process of recording the movements of objects or people.
  - Movements of one or more actors are sampled many times per second and translated into digital data.
  - Then the recording data is used for creating 2D or 3D character animation.
  - When it includes face and fingers or captures subtle expressions, it is often referred to as performance capture.

- The use of MoCap allows for more realistic and natural animations, as it captures the nuances of human motion.

Motion Capture

Methods for generating character animation

- There are several methods for generating character animation.
  - Key frame animation
  - Procedural animation
  - Mocap

- Before discussing the MoCap, let’s take a look at the other methods. Key frame animation.
  - This method involves defining specific poses, or keyframes, for a character at certain points in time.
  - Animators create these keyframes by manually adjusting the position and orientation of the character's joints or bones.
  - The computer then interpolates between these keyframes to generate the in-between frames, creating a smooth and continuous motion.
  - Keyframe animation allows for precise control over the character's movements and timing and is well-suited for animating non-humanoid characters, fantastical creatures, or highly stylized movements.

Motion Capture

Methods for generating character animation

- Key frame animation.

Motion Capture

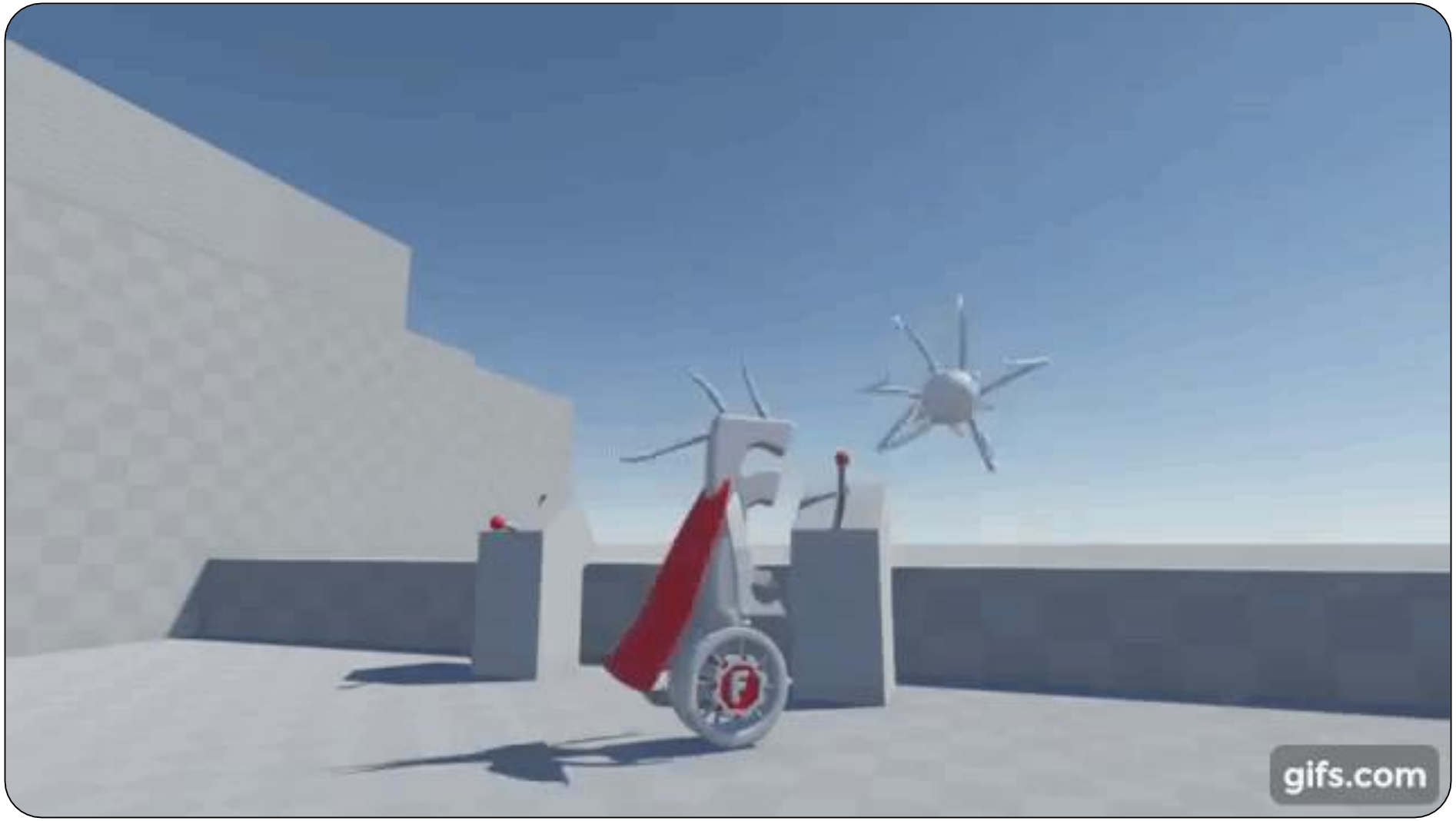
Methods for generating character animation

- Procedural animation
  - Procedural animation relies on algorithms, rules, or simulations to generate character movements, rather than manual keyframing.
  - This approach can involve physics-based simulations (e.g., cloth or fluid simulations), inverse kinematics (IK) for limb movement, or artificial intelligence algorithms to create lifelike behavior.
  - Procedural animation can save time and effort by automating complex or repetitive tasks, but it may require additional fine-tuning to achieve the desired results.

Motion Capture

Methods for generating character animation

- Procedural animation



Motion Capture

Advantages and disadvantages of MoCap

- Advantages
  - Realism: MoCap captures the nuances and subtleties of human movement, leading to more realistic and believable character animations. This is especially important when animating complex actions, such as athletic performances, dances, or fights.

  - Efficiency: Motion capture can save time and resources compared to keyframe animation. Once the motion data is captured, animators can quickly apply it to the digital characters, rather than creating the animations from scratch. This can lead to faster production timelines.

  - Consistency: MoCap can provide a consistent style and level of quality across multiple animations. As the movements are based on real-life performances, they can be more coherent and fluid than animations created through keyframing.

  - Performance-driven: Motion capture allows actors and performers to directly influence the animation, providing a stronger connection between the character and the performance. This can result in more expressive and emotionally engaging animations.

Motion Capture

Advantages and disadvantages of MoCap

- Disadvantages
  - Cost: MoCap systems, including cameras, suits, and software, can be expensive. Additionally, the capture process requires a dedicated space and specialized personnel, which can add to the overall cost.

  - Limited flexibility: Motion capture is ideal for realistic human or animal movements, but it may not be suitable for animating fantastical creatures, highly stylized characters, or non-humanoid characters. In these cases, keyframe animation or procedural techniques may be more appropriate.

  - Cleanup and refinement: MoCap data often requires significant post-processing, cleanup, and refinement before it can be used effectively. This can be time-consuming and may offset some of the efficiency gains of using motion capture.

  - Dependence on performer: The quality of the animation is heavily reliant on the skills and abilities of the performer. If the performance is not convincing, the resulting animation may also lack the desired impact.

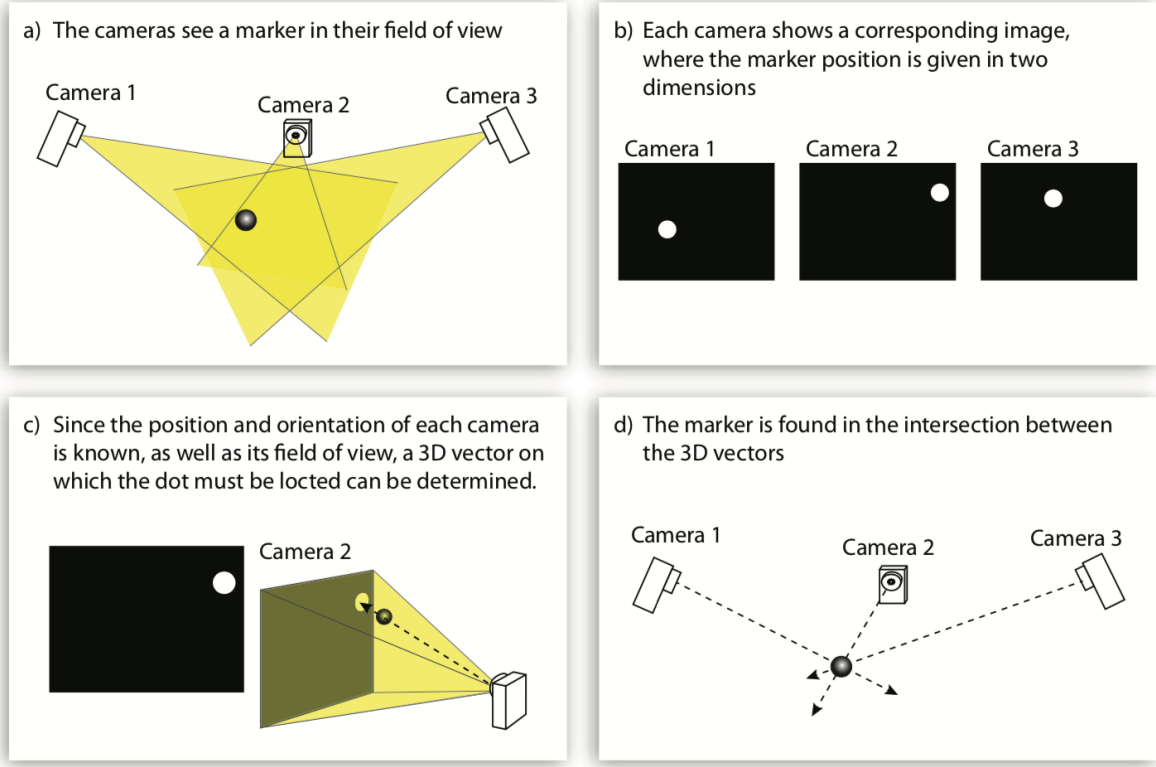
MoCap Methods and Systems

There are mainly five types of motion capture methods:

1. Optical motion capture (OMC)
   - This method involves tracking markers placed on an actor's body and capturing their movement using multiple cameras placed around the capture space.
   - The system triangulates the markers' positions in 3D space and generates a digital skeleton that can be applied to a character model.
   - There are mainly two types of optical motion capture: Passive and Active.

Triangulation[TRI]

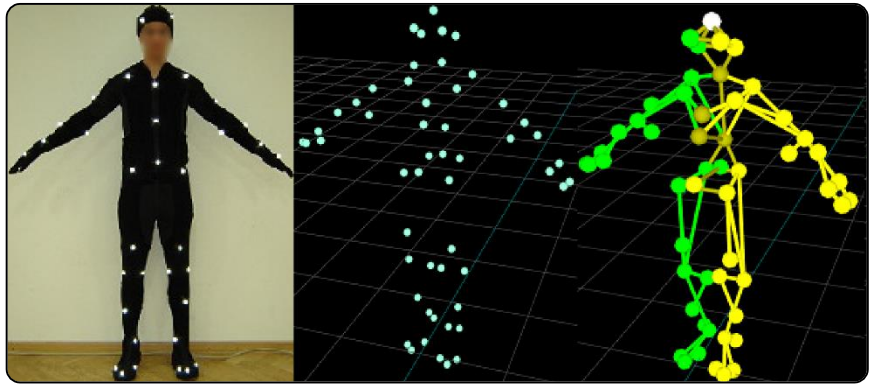
- Triangulation refers to the process of determining a point in 3D space given its 2D projections onto two, or more, images.
- The 2D position of a point from two or more cameras is overlapped and processed to reconstruct a common 3D point.



MoCap Methods and Systems

There are mainly five types of motion capture methods:

1. Optical motion capture (OMC)
    1) Passive optical motion capture:
        - The term “passive” is used because the markers in this system do not emit light or require any power source.
            - They rely on external light sources, such as infrared or visible light, to illuminate the them.
            - The markers are made of a retroreflective material, which passively reflects the light back towards its source, making them highly visible to the cameras.
            - The markers don’t have any built-in functionality and they simply reflect the light they receive.
        - Therefore, infrared or light cameras should be placed around the capture area to illuminate the markers.
            - When light hits the reflective markers, it bounces back towards the cameras, making the markers easy to detect and track.
            - The captured images are processed to identify the markers’ positions in each frame.
            - Then, the system triangulates the 3D positions of the markers based on the overlapping camera views and constructs a digital skeleton.



optical motion capture[OMC]

facial optical motion capture[MC]

MoCap Methods and Systems

There are mainly five types of motion capture methods:

1. Optical motion capture (OMC)
    2) Active optical motion capture:
        - The term “active” is used because the markers in this system actively emit light, typically in the form of infrared LEDs.
            - These markers require a power source and often have built-in circuitry to control their light emission.
            - The markers can be assigned unique IDs and turned on and off in a specific sequence, allowing the system to identify and track them individually.
        - Therefore, infrared cameras should be placed around the capture area to track the LED markers.
            - The motion capture software processes the captured images, identifies the markers based on their unique IDs, and tracks their positions in each frame.
            - Then, the system triangulates the 3D positions of the markers based on the overlapping camera views and constructs a digital skeleton.

active facial motion capture[AFMC]

active motion capture[AMC]

MoCap Methods and Systems

There are mainly five types of motion capture methods:

1. Optical motion capture (OMC)
   - Passive VS. Active
     - **Cost**: Passive systems are generally less expensive than active systems, as the markers do not require built-in LEDs or power sources.

     - **Marker Complexity**: Passive markers are lightweight and easy to attach to an actor’s body, making the setup process relatively straightforward. On the other hand, active markers are more complex than passive markers, as they require built-in LEDs, power sources, and sometimes wireless communication. This can make them larger, heavier, and potentially more cumbersome to attach to an actor's body.

     - **Lighting dependency**: The reflective markers can be highly visible under the right lighting conditions, making them easy for the cameras to detect and track. However, this can be affected by changes in lighting conditions or interference from other reflective surfaces in the capture area. On the other hand, active systems are less affected by those issues.

     - **Marker occlusion**: Active systems can often better handle occlusion issues, as the unique IDs allow the system to more easily recover the marker’s position when it reappears in the camera’s view.

MoCap Methods and Systems

There are mainly five types of motion capture methods:

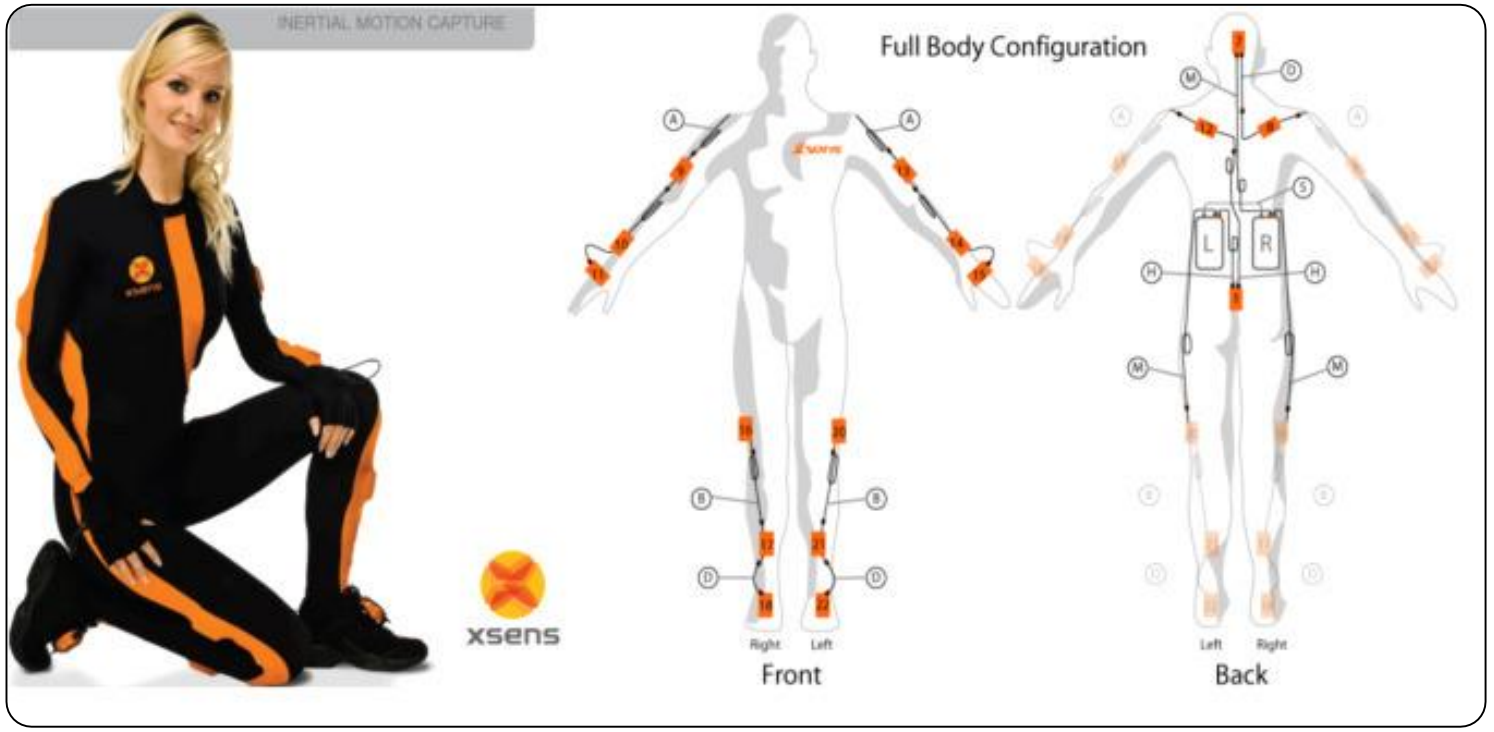
1. Optical motion capture (OMC)

optical motion capture system in USC[USC]

MoCap Methods and Systems

There are mainly five types of motion capture methods:

2. Inertial motion capture (IMC)
   - This method uses small, wearable sensors (Inertial Measurement Units – IMUs) placed on an actor's body to capture movement data.
   - The sensors measure acceleration (using accelerometers), angular velocity (using gyroscopes), and magnetic field orientation (using magnetometers), which are then processed to determine the actor's position and orientation.



MoCap Methods and Systems

There are mainly five types of motion capture methods:

2. Inertial motion capture (IMC)
   - Advantages:
     - **Portability and flexibility**: IMC systems are more portable and flexible than OMC systems, as they do not require cameras or a dedicated capture space. This allows motion capture to be performed in various environments and over larger areas.
     - **No occlusion issues**: Since IMC systems use sensors directly attached to the actor's body, they do not suffer from occlusion problems, which can be an issue with OMC systems when markers are not visible to the cameras.
     - **Faster setup**: IMC systems generally have a faster and easier setup process, as they do not require the precise placement and calibration of cameras, nor the need for specialized lighting.
     - **Real-time feedback**: Many IMC systems can provide real-time motion data, allowing for immediate feedback during the capture session.

MoCap Methods and Systems

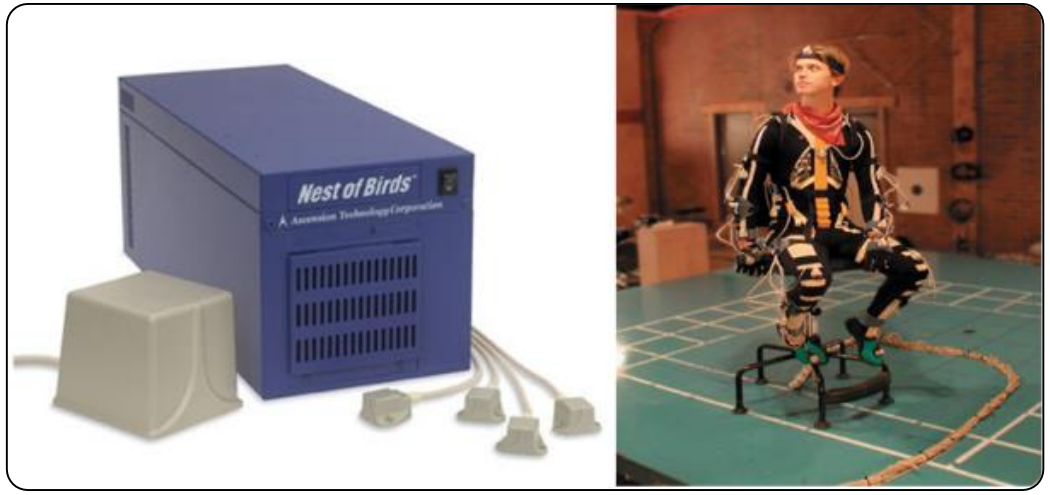
There are mainly five types of motion capture methods:

2. Inertial motion capture (IMC)
   - Disadvantages:
     - **Positional accuracy**: IMC systems may have lower positional accuracy compared to OMC systems, due to the integration errors and potential sensor drift that can occur when calculating position from acceleration data.
     - **Magnetic interference**: The performance of IMC systems can be affected by magnetic interference from metal objects or electronic devices, which can impact the accuracy of the magnetometer readings used for orientation calculations.
     - **Battery life**: IMUs rely on batteries, which may require charging or replacement during long motion capture sessions.

MoCap Methods and Systems

There are mainly five types of motion capture methods:

3. Magnetic motion capture
   - This method relies on magnetic fields to track an actor’s movements.
   - Sensors placed on the actor's body detect changes in the magnetic field generated by a transmitter, and these changes are translated into 3D positional data.
   - While magnetic motion capture is less susceptible to occlusion than optical systems, it can be affected by interference from metal objects or other magnetic fields.
4. Mechanical motion capture
   - This method uses a rigid exoskeleton worn by the actor to capture movement data.
   - The exoskeleton's joints have sensors that record their angles, providing data for the character's joint angles.
   - Mechanical motion capture is not as popular as other methods due to its limited range of motion and the discomfort of wearing an exoskeleton.

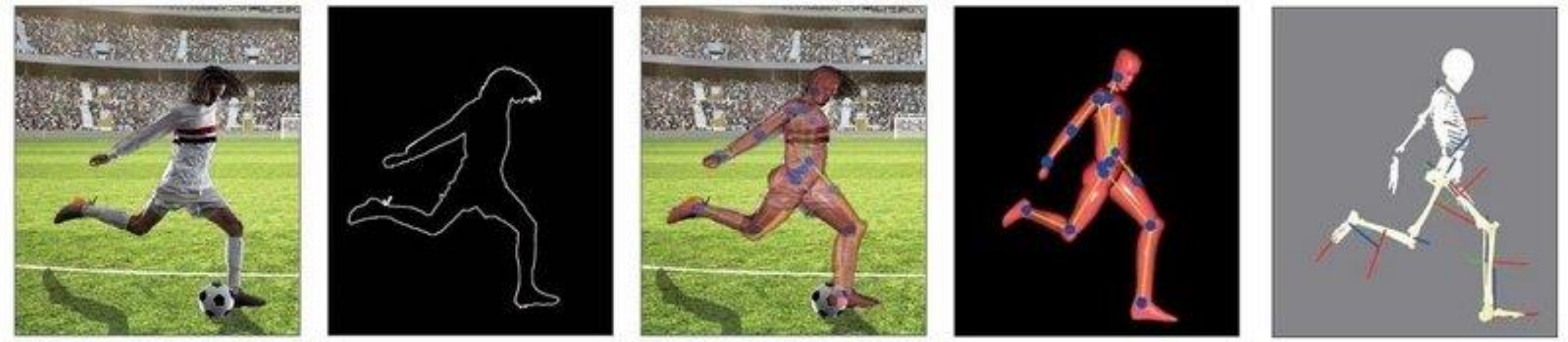


magnetic & mechanical motion capture[MMC]

MoCap Methods and Systems

There are mainly five types of motion capture methods:

5. Markerless Motion Capture
   - This method uses computer vision algorithms to analyze the movements of an actor without requiring markers or sensors on their body.
   - It can be achieved using depth-sensing cameras, machine learning, or a combination of both.
   - Markerless motion capture is still under development but has seen significant advancements in recent years.



markerless motion capture[MLMC]

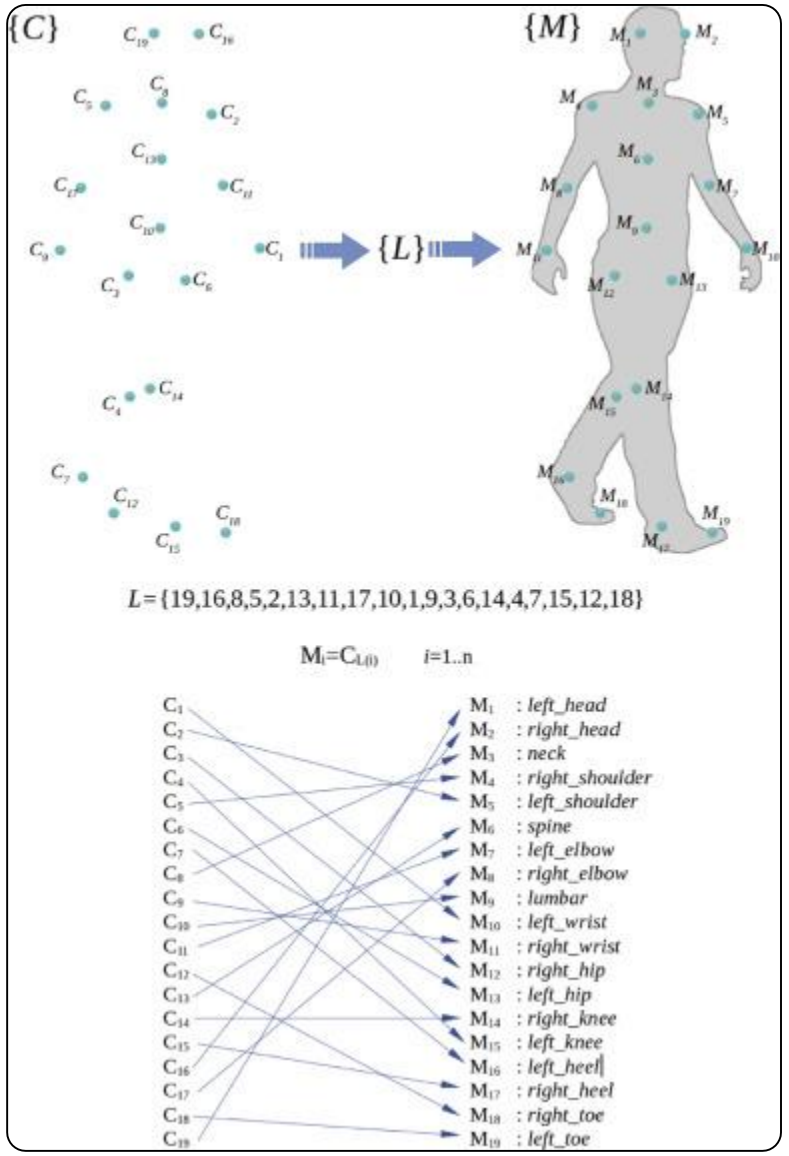
Post-processing of POMC

Why does passive optical motion capture (POMC) require post-processing?

- POMC is traditionally been more widely used than AOMC because it is less expensive, simpler to use, and familiar to end users (there are many experts, well-known workflows, and many examples).

- POMC requires post-processing to ensure accuracy, and improve the data quality.
  - **Resolving marker ambiguity (marker labeling)**: In POMC, markers may look identical, and during fast or complex movements, the system can have difficulty distinguishing between them. As a result, markers may be incorrectly assigned or swapped. Post-processing involves reviewing the data and correcting any marker assignment errors.

  - **Reducing noise**: The raw motion capture data may contain noise, which can be caused by various factors such as camera resolution, lighting conditions, or marker reflections. Post-processing can involve applying noise reduction techniques, such as filtering or smoothing algorithms, to reduce jitter and improve the overall quality of the motion data.



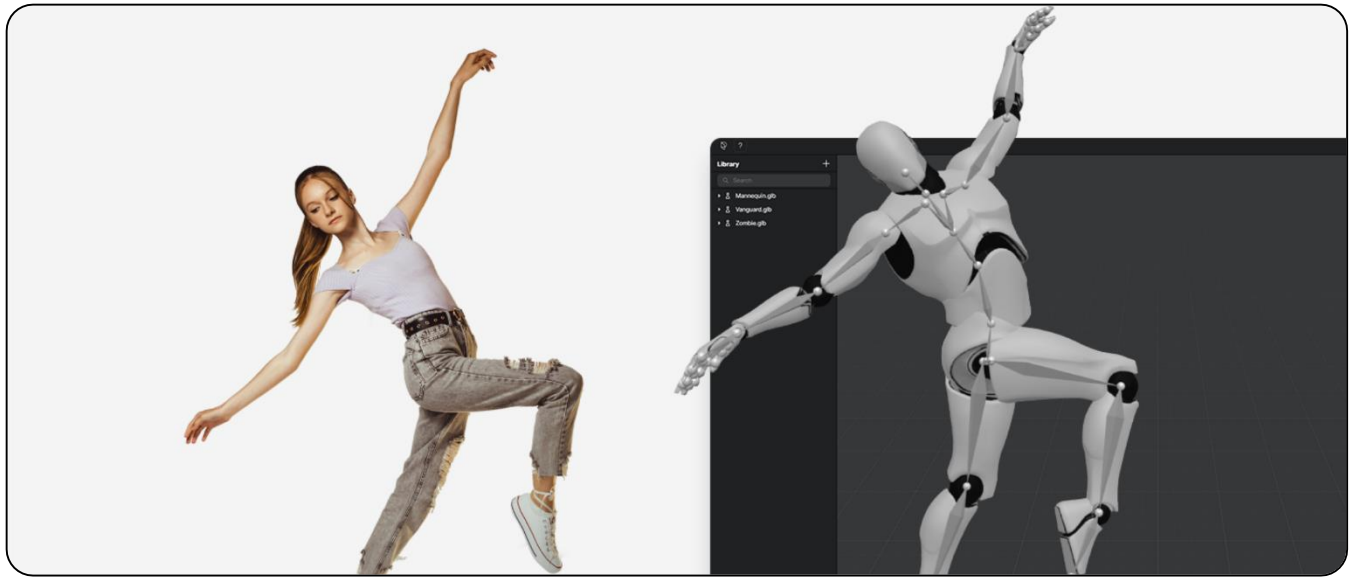
marker labeling [MALB]

Post-processing of POMC

Why does passive optical motion capture (POMC) require post-processing?

- (continued) POMC requires post-processing to ensure accuracy, and improve the data quality.
  - **Gap filling**: If a marker is occluded (not visible to at least two cameras), its 3D position cannot be accurately calculated, resulting in gaps in the data. During post-processing, these gaps can be filled using interpolation techniques or by manually adjusting the marker positions to create a continuous motion.

  - **Rigging and retargeting**: To use the motion capture data in animation, the data needs to be mapped onto a digital skeleton that represents the character or object being animated. This involves rigging, which is the process of creating a digital skeleton with joints and bones that correspond to the actor's body, and retargeting, which involves mapping the motion data from the markers onto the corresponding joints and bones of the digital skeleton.



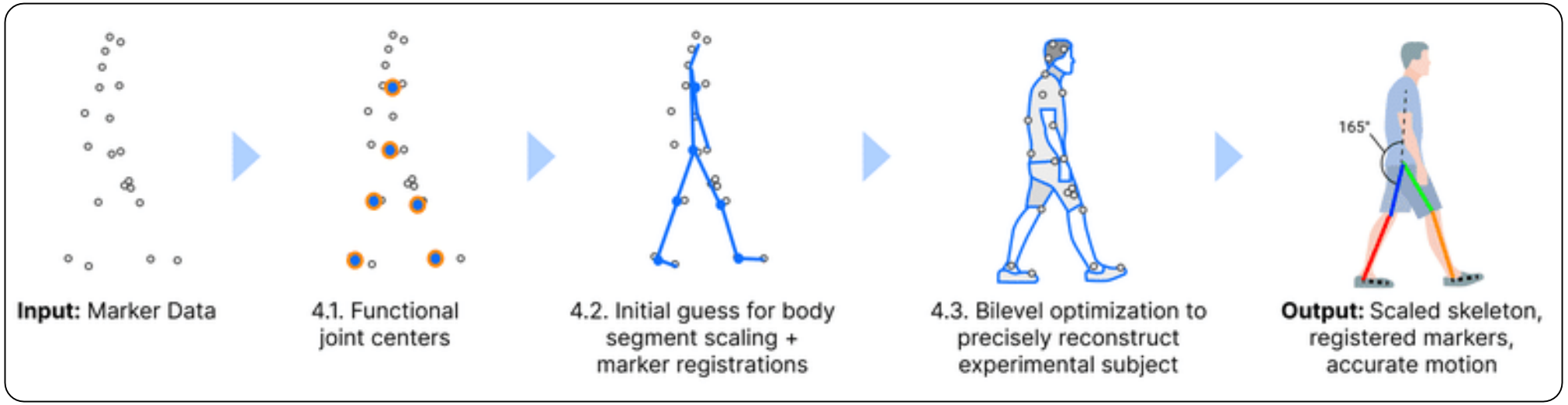
rigging and retargeting[RAT]

Post-processing of POMC

Why does passive optical motion capture (POMC) require post-processing?

- (continued) POMC requires post-processing to ensure accuracy, and improve the data quality.
  - **Constraint enforcement**: Sometimes, it's necessary to enforce constraints on the motion data to ensure that it accurately represents the intended movement and follows the physical limitations of the character being animated. For example, constraints can be applied to maintain a consistent distance between two markers, limit the range of motion of a joint, or ensure that a character's feet stay on the ground during a walk cycle.

  - **Cleanup and optimization**: After the initial post-processing steps, the motion data may require further cleanup and optimization to ensure that the final animation looks smooth and natural. This can involve adjusting the timing of keyframes, fine-tuning the position and orientation of joints, or adding secondary animations to enhance the overall performance.



data optimization[OPT]

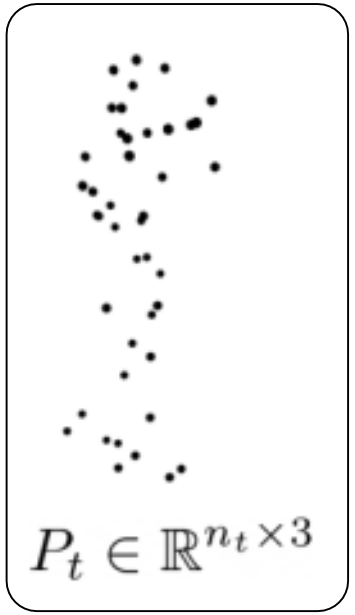
Mocap Labeling Problem

Correctly assigning labels to the markers in motion capture data

- Due to occlusion, fast or complex movements, or visual ambiguity, POMC systems can have difficulty distinguishing between markers.
  - Even worse, markers often look identical in POMC because they have no unique IDs.
  - This problem can lead to errors in marker assignments, causing the motion data to be misinterpreted.

- To resolve this problem, manual or automatic labeling methods can be used.
  - One recent method is “SOMA: Solving Optical Marker-Based MoCap Automatically, 2021 CVPR”.
  - Let’s take a look at the method.

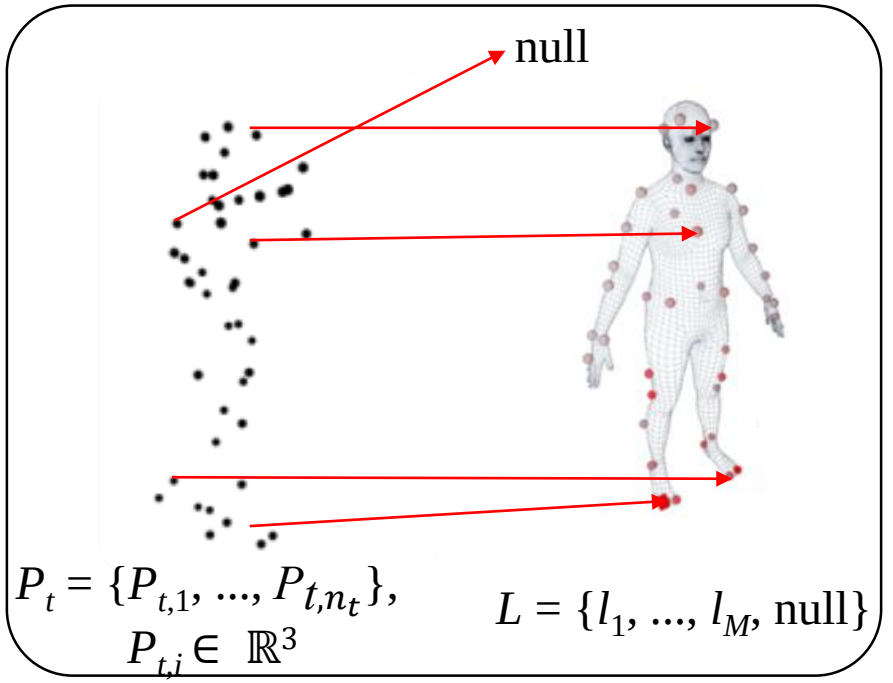
- A mocap point cloud (MPC) is a time sequence with T frames of 3D points:
  - MPC = $\{P_1, ..., P_T\}$,
  - $P_t = \{ P_{t,1}, ..., P_{t,n_t} \}$, $P_{t,i} \in \mathbb{R}^3$,
    where $|P_t| = n_t$, for each time step $t \in \{1{:}T\}$ and n_t = the number of input data at frame t.



SOMA: Mocap Labeling Problem

Correctly assigning labels to the markers in mocap data

- The goal of mocap labeling is to assign each point (or tracklet) to a corresponding marker label:
  - $L = \{l_1, ..., l_M, \text{null}\}$,
  - The set of marker labels include an extra null label for points that are not valid markers (ghost marker, noise marker), hence $|L| = M + 1$.
  - Valid point labels and tracklets of them are subject to three constraints:
    - C_1: Each point $P_{t,i}$ can be assigned to at most one label and vice versa.
    - C_2: Each point $P_{t,i}$ can be assigned to at most one tracklet.
    - C_3: The label null is an exception that can be matched to more than one point and can be present in multiple tracklets in each frame.

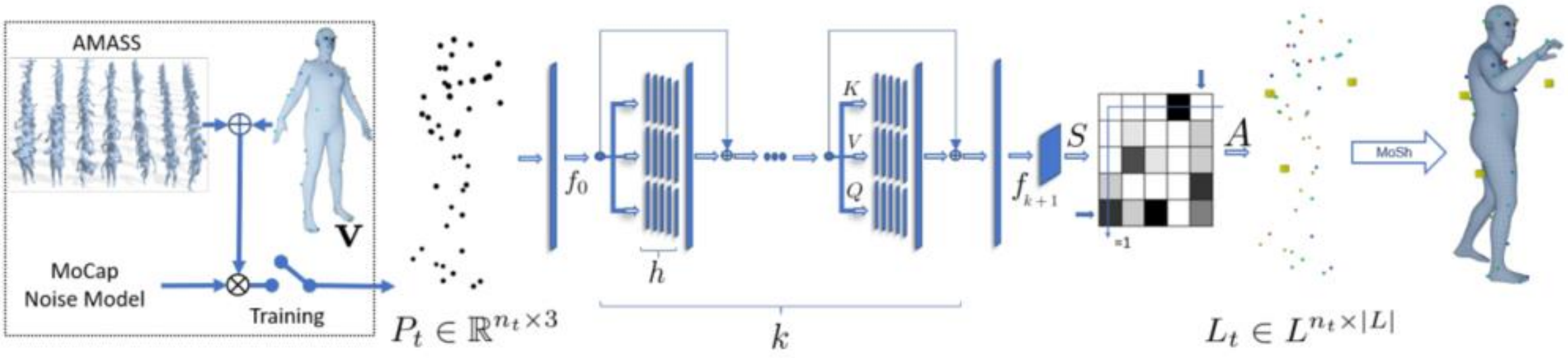


SOMA: Self-Attention on MPC

Correctly assigning labels to the markers in mocap data

- The SOMA system pipeline is shown in the below:
  - The input to SOMA is a single frame of sparse, unordered, points, with cardinality varying for each timestamp due to occlusions and ghost points.
    - Cardinality refers to the number of elements in a set.

  - To process such data, SOMA exploit multiple layers of self-attention.

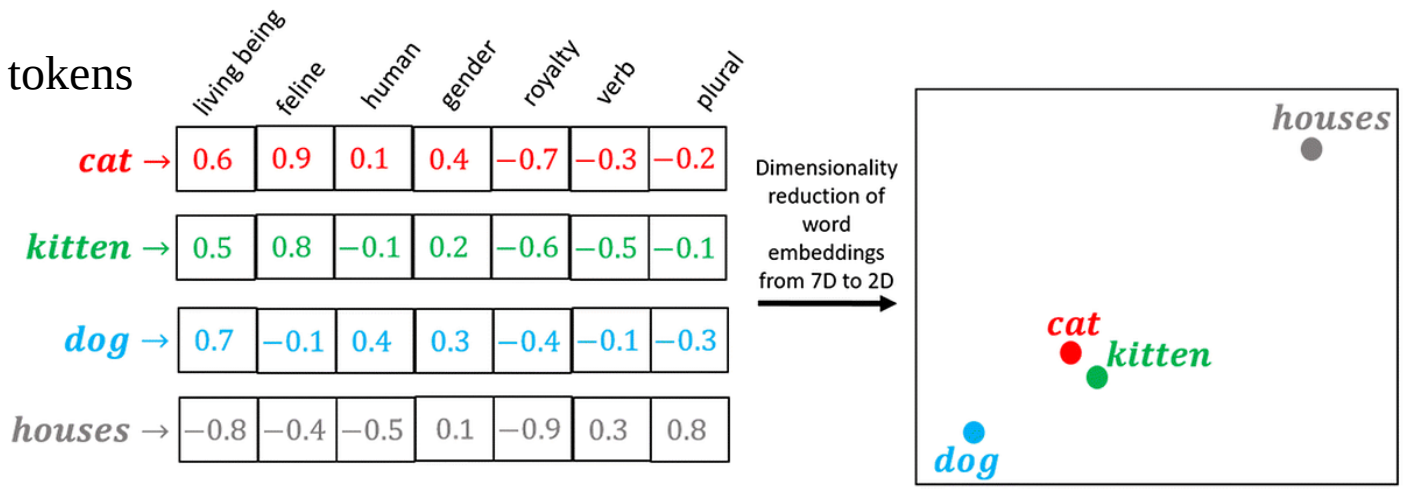


SOMA: Self-Attention on MPC

What is Attention?

- In artificial neural networks, attention is a technique that is meant to mimic cognitive attention.
  - The effect enhances some parts of the input data while diminishing other parts the motivation being that the network should devote more focus to the small, but important, parts of the data.
  - Learning which part of the data is more important than another depends on the context, and this is trained by gradient descent.

- Given a sequence of tokens t_i, labeled by the index i, a neural network computes a soft weight w_i for each t_i with the property that w_i is non-negative and $\sum_i w_i = 1$.
  - Each t_i is assigned a value vector v_i which is computed from the word embedding of the ith token.



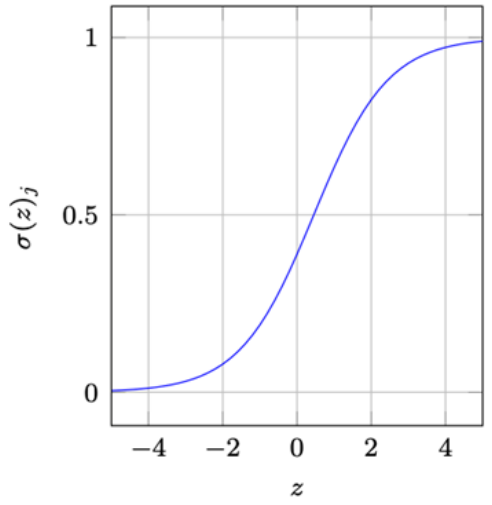
word embedding example ($d = 7$)[WEE]

	I	love	you
je	0.94	0.02	0.04
t'	0.11	0.01	0.88
aime	0.03	0.95	0.02

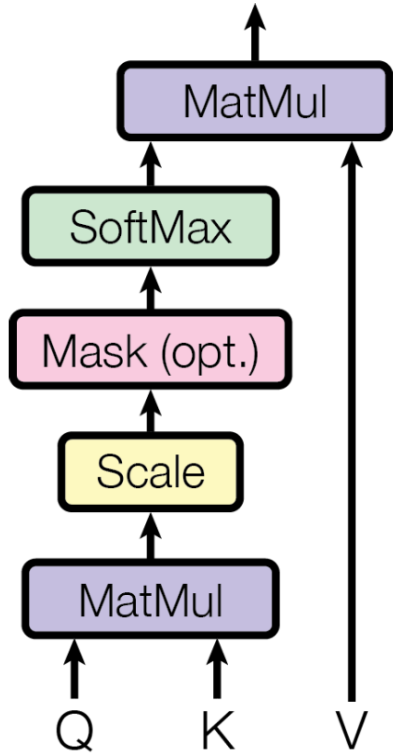
SOMA: Self-Attention on MPC

What is Attention[AIAN]?

- Given embedded vectors obtained from tokens, the weighted average $\sum_i w_i v_i$ is the output of the Attention.
  - To compute weights, the query-key mechanism is used.
  - We can calculate a Query's Attention by using Query (Q), Key (K), and Value (V) by using:
    - Query's Attention (Q, K, V) = softmax $(\frac{QK^T}{\sqrt{d_k}})$V
    - Q: Current token.
    - K: The token we want to obtain attention.
    - V: The token we want to obtain attention (initially identical to K).
    - d_k: Embeded vector dimension



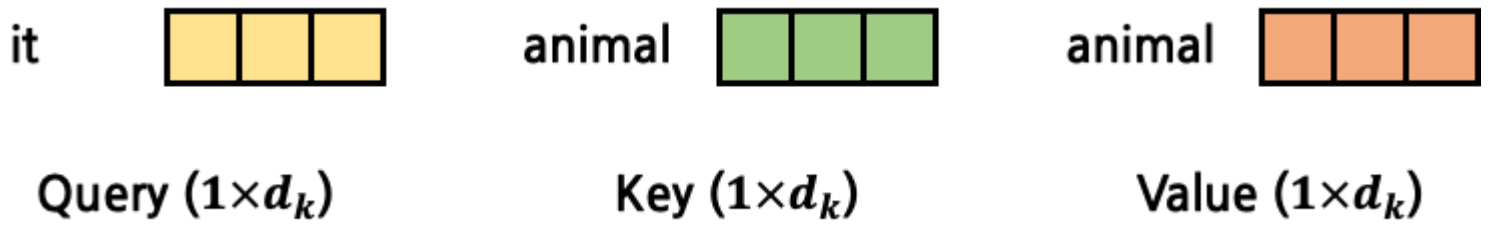
Softmax activation function.



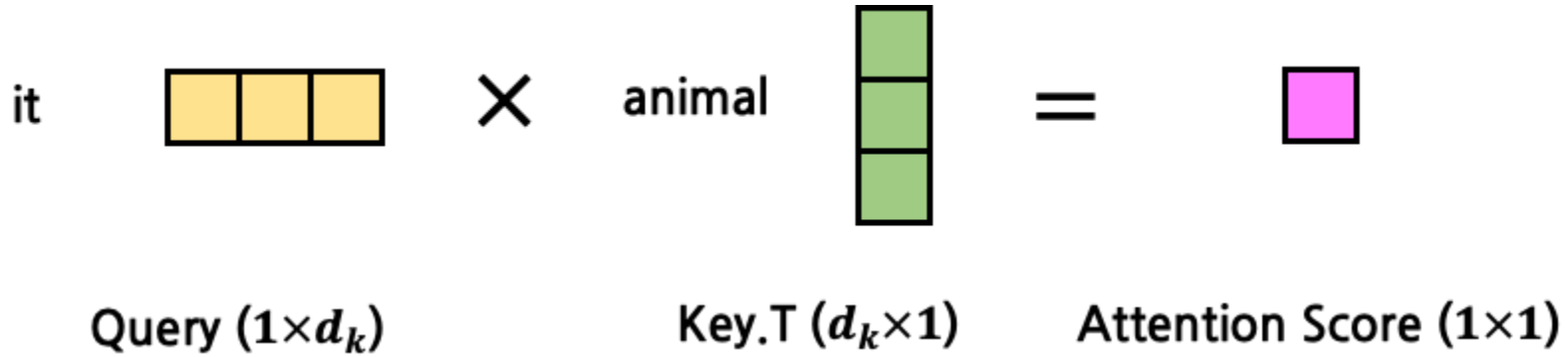
SOMA: Self-Attention on MPC

What is Attention?

- Let's see the example[TF].
  - We have "The animal didn't cross the street, because it was too tired".
  - Let's compute the attention of query 'it' with respect to 'animal'. The embedded vectors are as follows (d_k = 3):



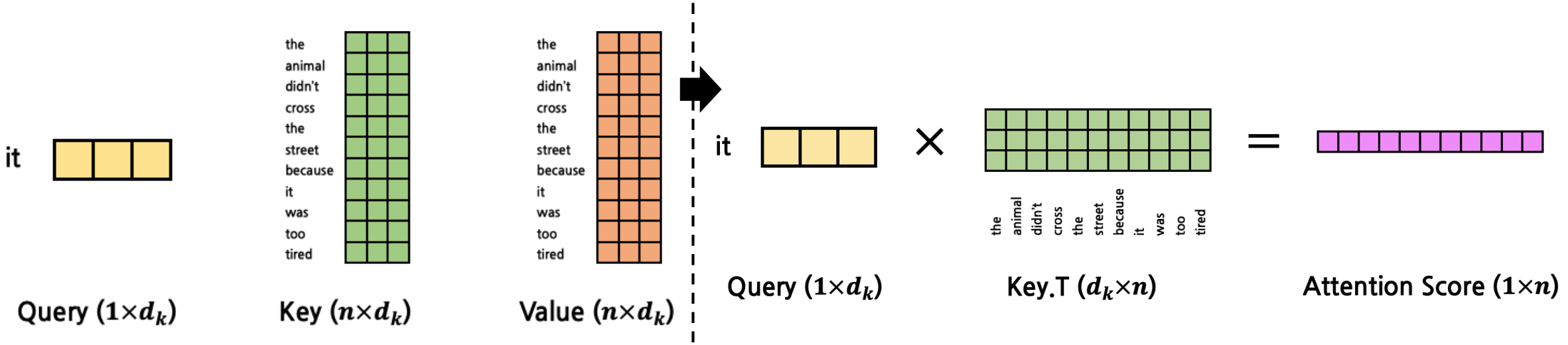
  - Computing QK^T is to obtain 'Attention Score' (recall that Query's Attention (Q, K, V) = softmax $(\frac{QK^T}{\sqrt{d_k}})$V).
    - To avoid gradient vanishing due to the too large value, scaling is conducted by dividing $\sqrt{d_k}$.



SOMA: Self-Attention on MPC

What is Attention?

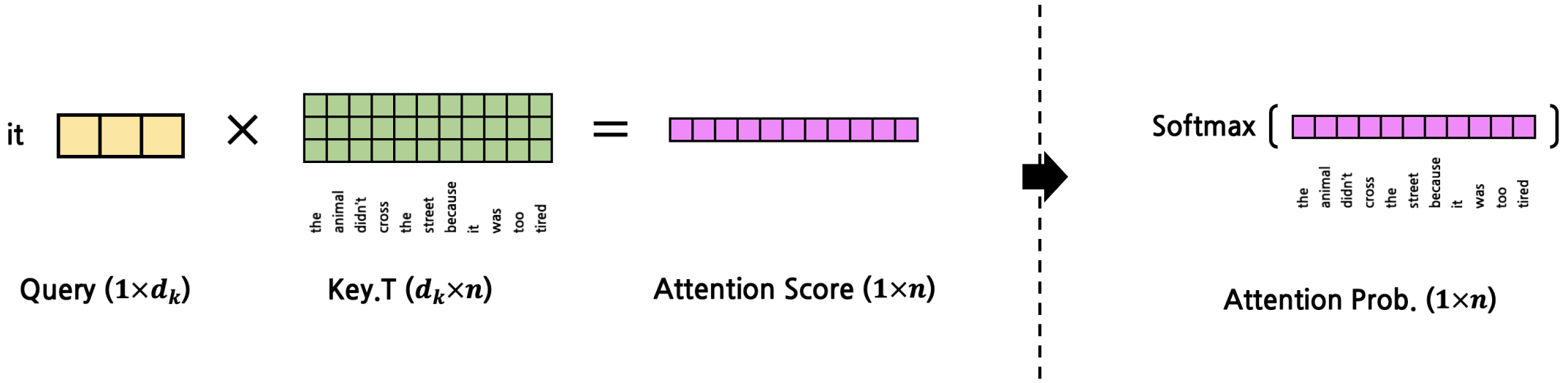
- Let's see the example[TF].
  - So far, we have calculated 1:1 Attention. However, the goal is to find the highest Attention value in a given set of tokens for a given Query.
    - For example, given "The animal didn't cross the street, because it was too tired", our goal is to find the token ("The", "animal", ... , "tired") with the highest Attention value with respect to "it".
  - Therefore, we need to compute 1:*N* Attention.
    - Here, Q = "it"; K and V = entire tokens in the given sentence.



SOMA: Self-Attention on MPC

What is Attention?

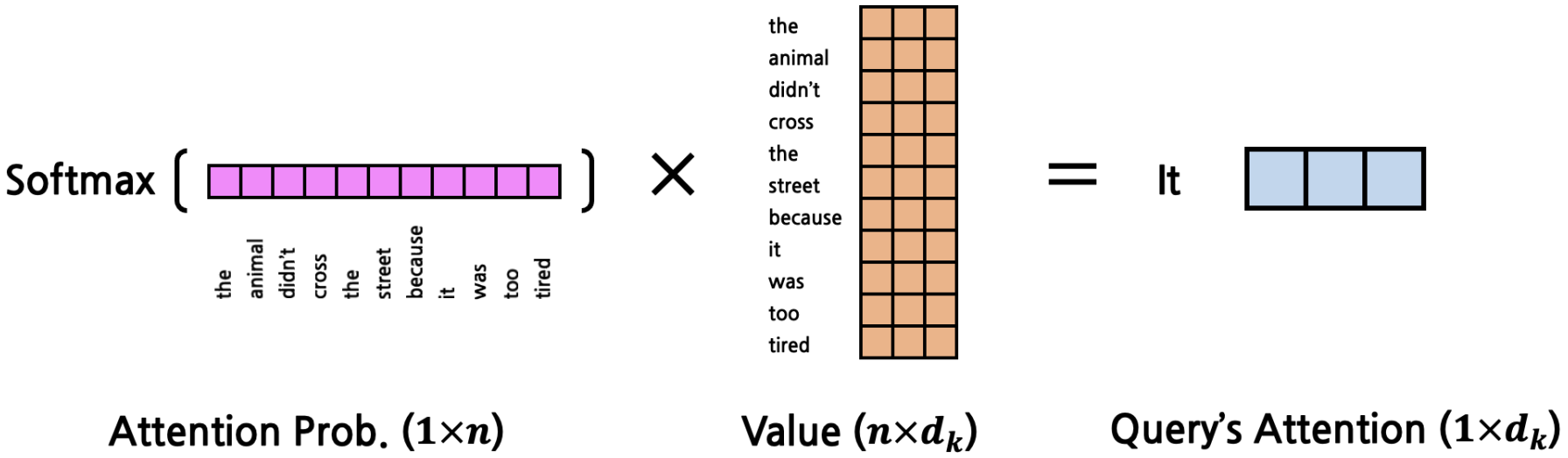
- Let's see the example[TF].
  - Given Attention Score, let's apply softmax() to obtain Attention Probability.
    - Note that the sum of all elements of Attention Probability is 1.
    - The value of each element represents how the given Q is closely related to the given K.



SOMA: Self-Attention on MPC

What is Attention?

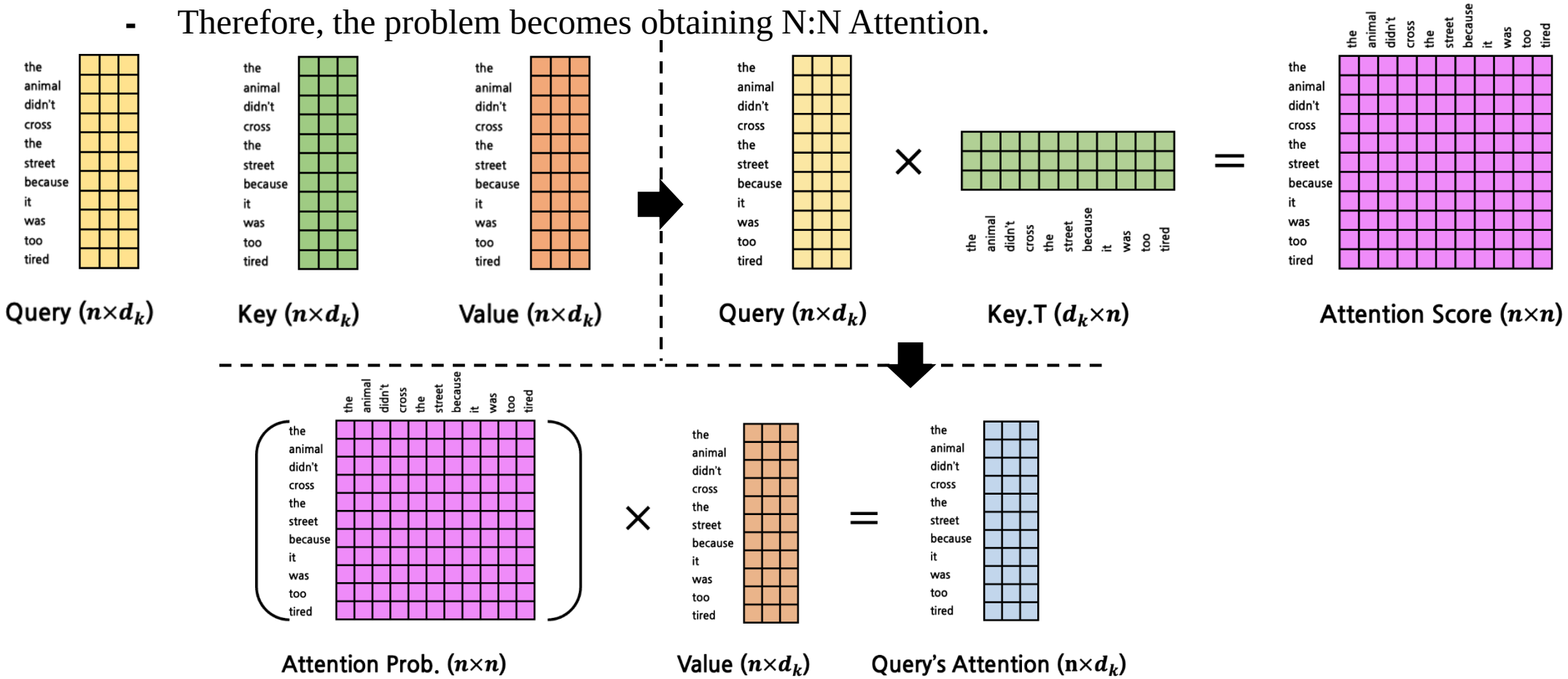
- Let's see the example[TF].
  - Afterward, the obtained Attention Probability is multiplied with V.
    - This means that obtaining a unified vector considering the set of attention probabilities.



SOMA: Self-Attention on MPC

What is Attention?

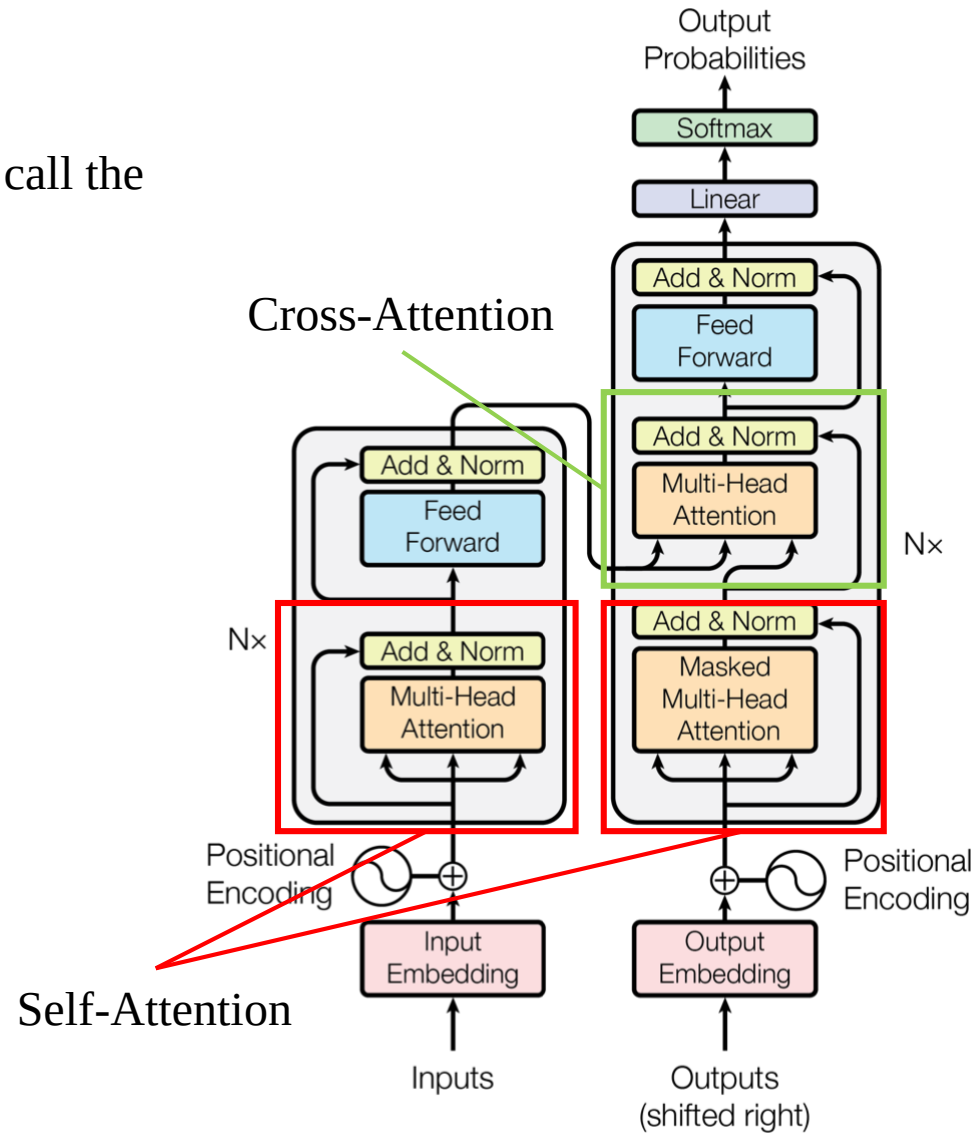
- Let's see the example[TF].
  - So far, we have calculated 1:N Attention ("it" ↔ entire sentence). However, our goal is to obtain Attention for all tokens.
    - Therefore, the problem becomes obtaining N:N Attention.



SOMA: Self-Attention on MPC

What is Attention?

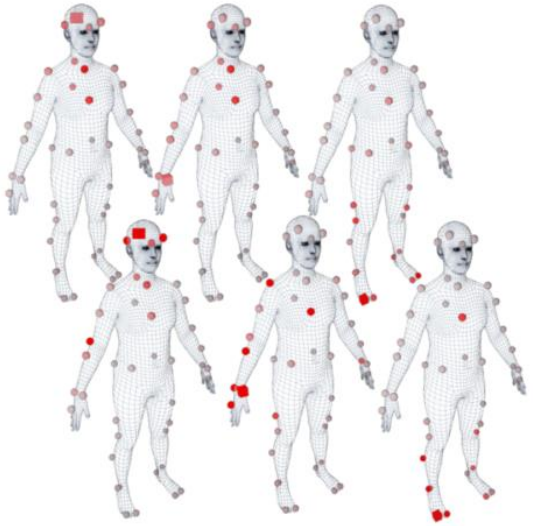
- Self attention?
  - When all Q, K, V are originated from the same data, we call the result Attention Self-Attention.

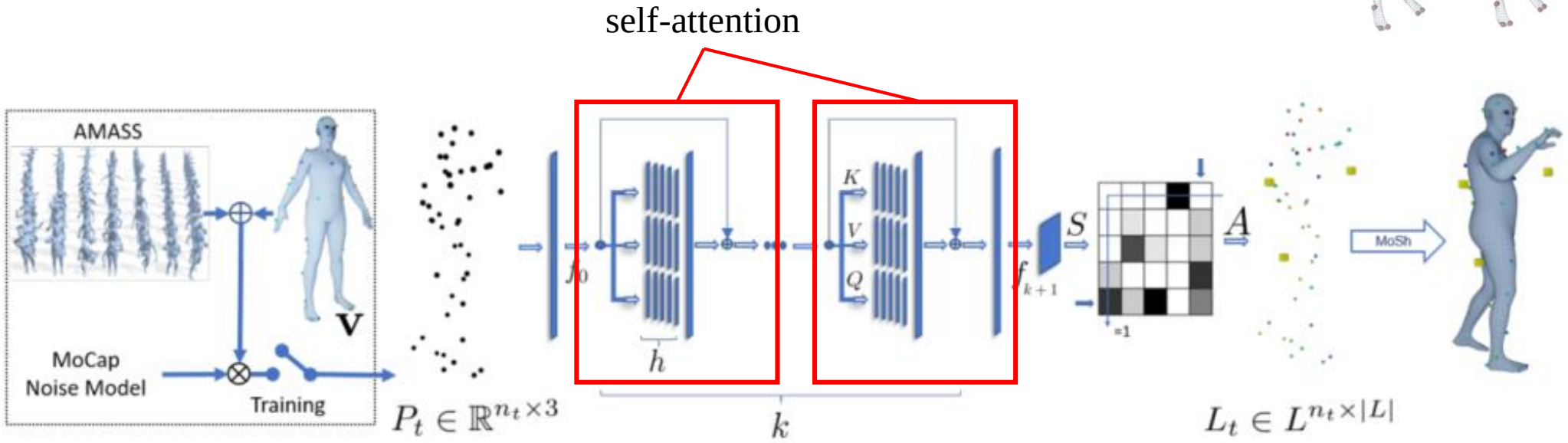


SOMA: Self-Attention on MPC

Correctly assigning labels to the markers in mocap data

- SOMA exploit multilayered multi-head self-attention.
  - Self-attention span is defined as the average of attention weights over random sequences picked from the validation dataset.
  - In the right figure, the intensity of the red color correlates with the amount of attention.
  - As shown in the figure, the self-attention network has figured out the spatial structure of the body and correlations between parts.
  - Self-attention layer is applied *k* times and outputs the attention probabilities.

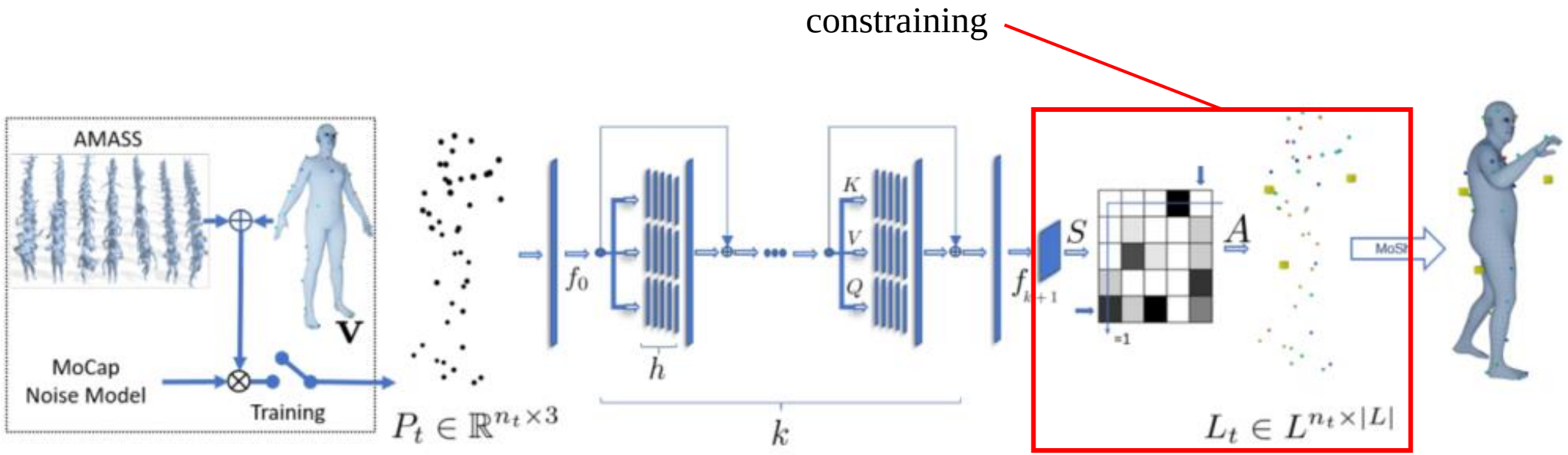




SOMA: Constraining

Correctly assigning labels to the markers in mocap data

- Constrained point labeling (this lecture does not address this deeply).
  - In the final stage, SOMA predicts a non-square matrix S $\in \mathbb{R}^{n_t \times M}$ (recall that $L = \{l_1, ..., l_M, \text{null}\}$).
    - To satisfy the constraints C_1 and C_2, SOMA employs a log-domain, stable implementation of optimal transport.
    - The optimal transport layer depends on iterative Sinkhorn normalization, which constraints rows and columns to sum to 1 for available points and labels.
    - To deal with massing markers and ghost points, SOMA introduces *dustbins* by appending an extra last row and column to the score matrix.
  - After normalization, the augmented assignment matrix, A` $\in [0,1]^{(n_t+1)\times|L|}$.
  - By dropping the appended row, for unmatched labels, the final normalized matrix A $\in [0,1]^{n_t \times |L|}$ is obtained.

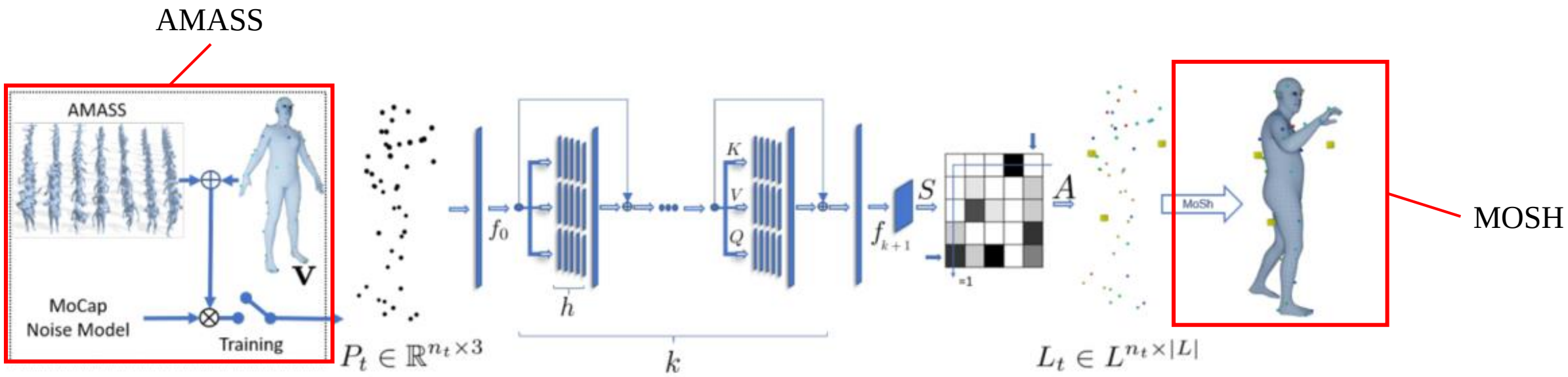


SOMA: AMASS and MoSh

Correctly assigning labels to the markers in mocap data

- AMASS and MoSh.
  - To learn a robust model, SOMA exploits AMASS[AMASS].
    - The authors generate AMASS motions for 3664 bodies.
    - Given ground truth data, authors corrupt them with various controllable noise sources.

  - To fit a skeletal model to the labeled markers, SOMA exploits MoSh[Mosh].
    - MoSh gives an animated body with a skeletal structure, yielding a full 3D body model.



Reference

[MC] https://en.wikipedia.org/wiki/Motion_capture
[TRI] https://www.futurelearn.com/info/courses/motion-capture-course/0/steps/272015
[OMC] Skurowski, P., & Pawlyta, M. (2021). Gap reconstruction in optical motion capture sequences using neural networks. Sensors, 21(18), 6115.
[AFMC] http://www.phasespace.com/clients_motioncapture_nyc.html
[AMC] https://sentimentalflow.wordpress.com/2017/01/30/first-blog-post/
[USC] https://www.cgarena.com/newsworld/vicon-usc-motion-capture.php
[IMC] Ameli, S., Naghdy, F., Stirling, D., Naghdy, G., & Aghmesheh, M. (2017). Objective clinical gait analysis using inertial sensors and six minute walking test. Pattern Recognition, 63, 246-257.
[MMC] Gmiterko, A., & Lipták, T. (2013). Motion capture of human for interaction with service robot. American Journal of Mechanical Engineering, 1(7), 212-216.
[MLMC] http://www.simi.com/en/products/movement-analysis/markerless-motion-capture.html
[MALB] Bascones, J. J., Graña, M., & Lopez-Guede, J. M. (2019). Robust labeling of human motion markers in the presence of occlusions. Neurocomputing, 353, 96-105.
[RAT] https://www.firstperson.is/1p-labs/motion-capture
[OPT] Werling, K., Raitor, M., Stingel, J., Hicks, J. L., Collins, S., Delp, S., & Liu, C. K. (2022). Rapid bilevel optimization to concurrently solve musculoskeletal scaling, marker registration, and inverse kinematic problems for human motion reconstruction. bioRxiv, 2022-08.
[WEE] https://medium.com/@hari4om/word-embedding-d816f643140
[TF] https://cpm0722.github.io/pytorch-implementation/transformer
[AIAN] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. Advances in neural information processing systems, 30.
[AMASS] Mahmood, N., Ghorbani, N., Troje, N. F., Pons-Moll, G., & Black, M. J. (2019). AMASS: Archive of motion capture as surface shapes. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 5442-5451).
[MOSH] Loper, M., Mahmood, N., & Black, M. J. (2014). MoSh: motion and shape capture from sparse markers. ACM Trans. Graph., 33(6), 220-1.