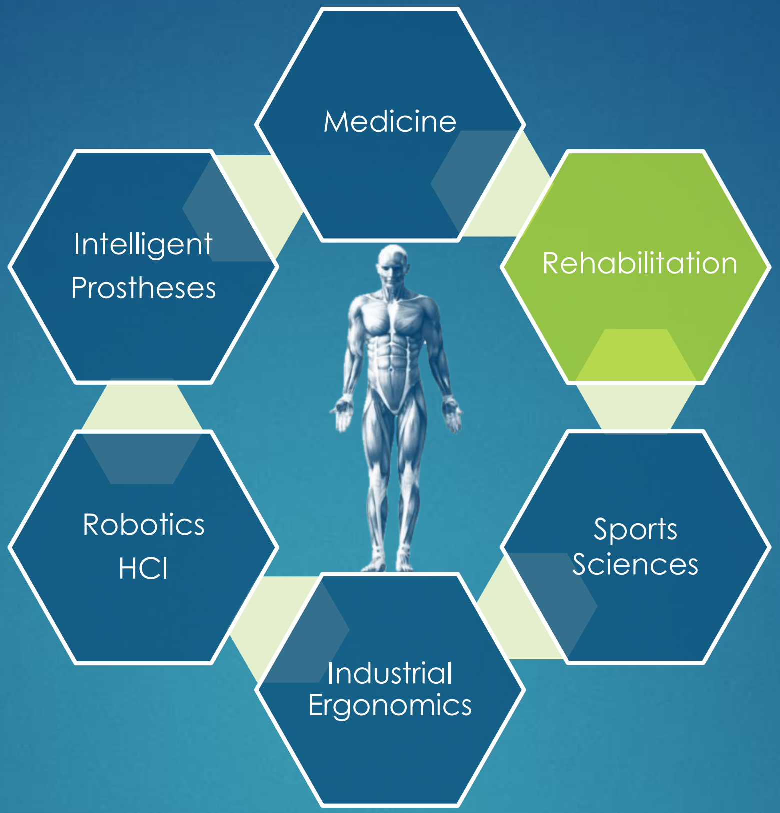
wearHEALTH



Towards Inertial Musculoskeletal Analysis:

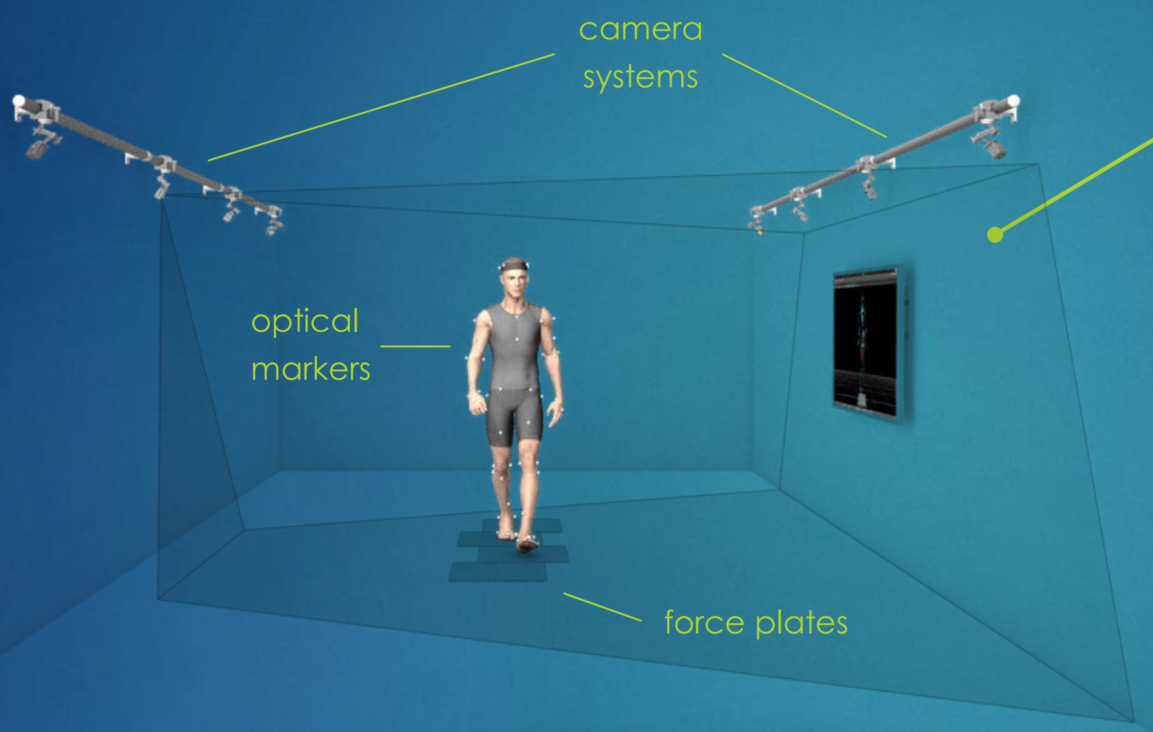
Effects of Sensor-to-Segment Calibration on Predicted Ground Reaction Forces

---

MASTER‘S THESIS – Felix Laufer

wearHEALTH, TU Kaiserslautern

Motivation – Musculoskeletal Analysis



typical gait lab equipment

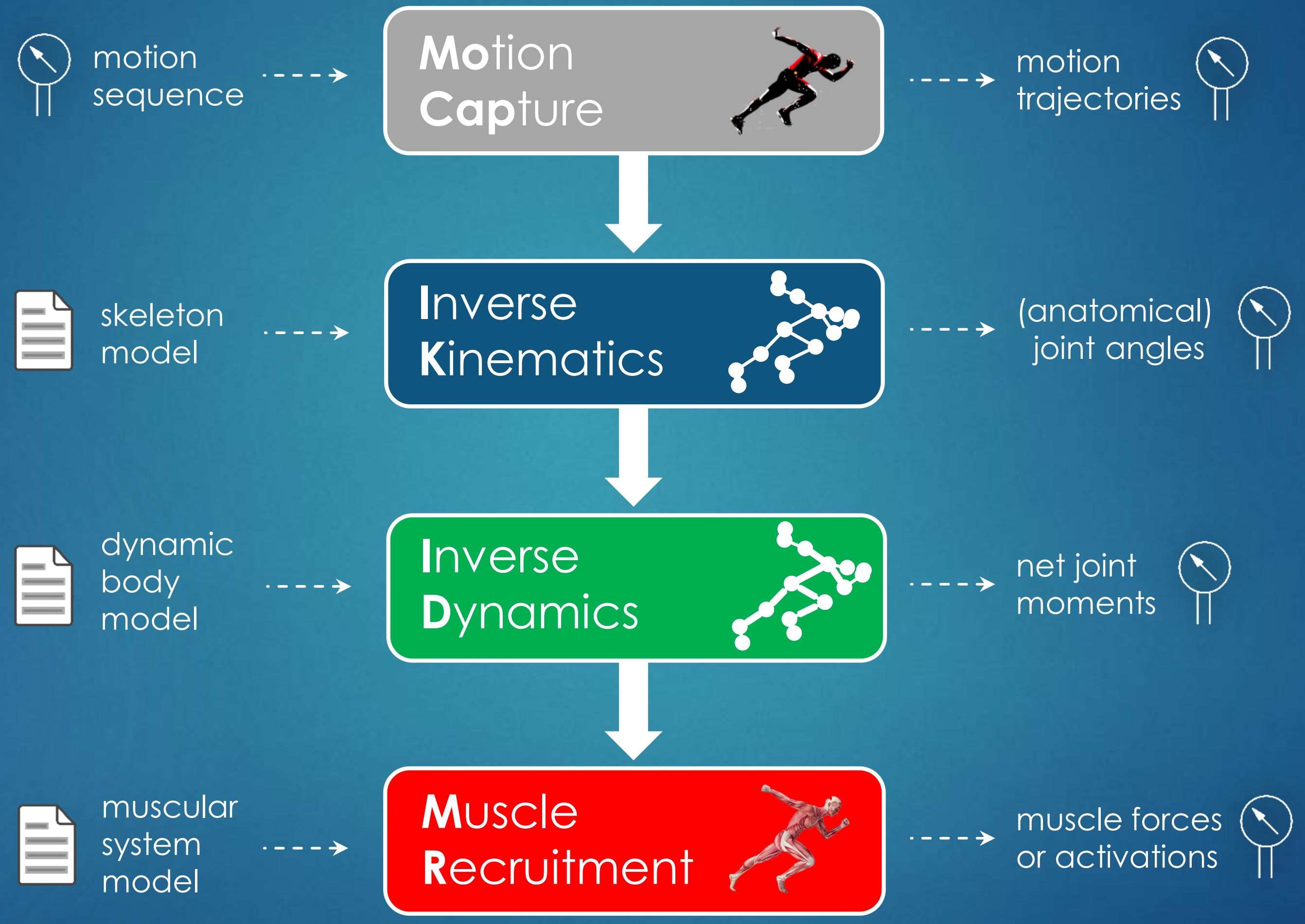
(Clinical) Motion and Gait Analysis

kinematic & kinetic motion parameters:

- anatomical kinematics
- ground reaction forces
- joint reaction forces
- muscle forces / activations

long-term goal:
flexibel & mobile body-worn system
➔ inertial musculoskeletal analysis

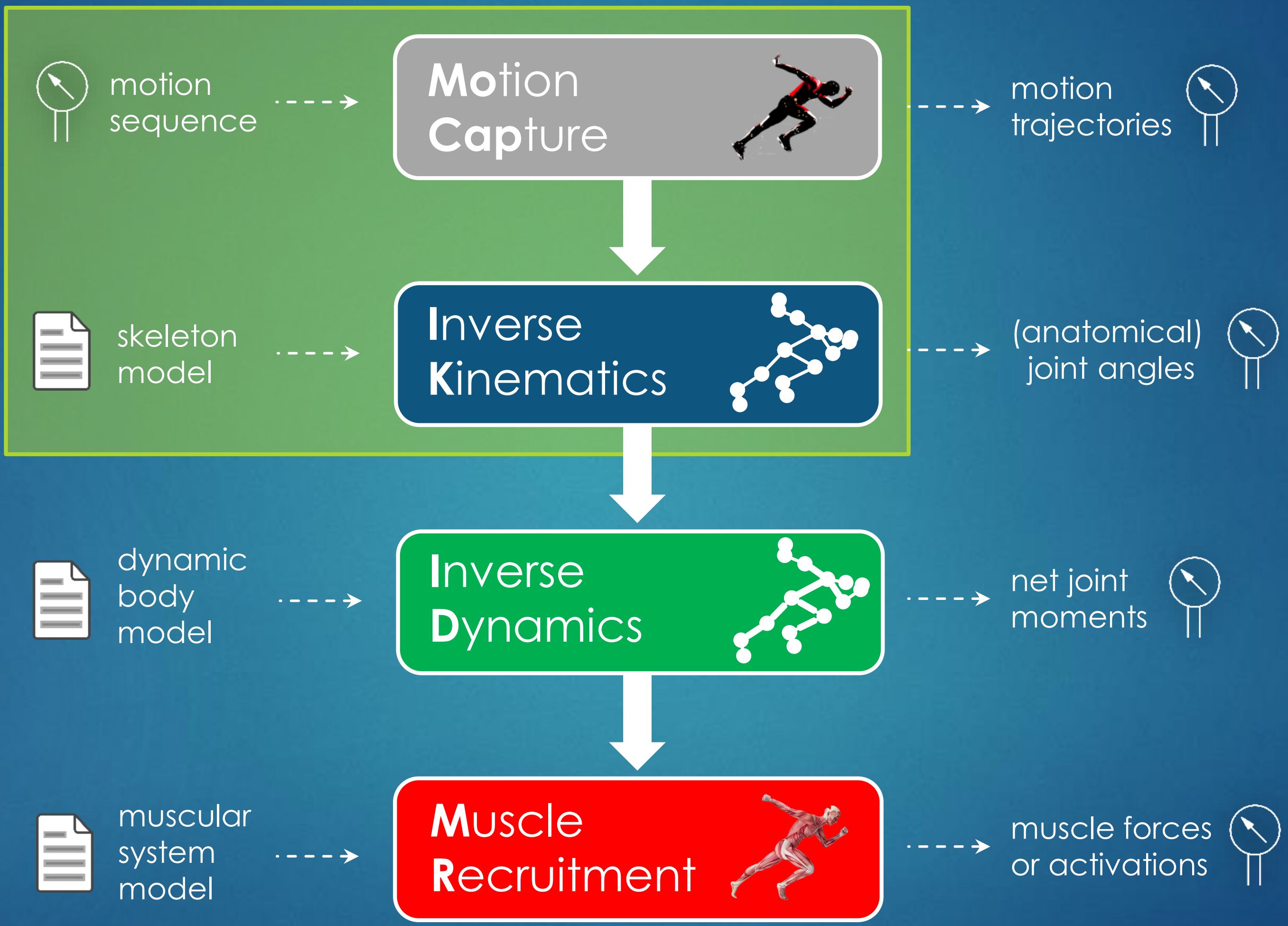
Introduction – Musculoskeletal Analysis Pipeline



Agenda

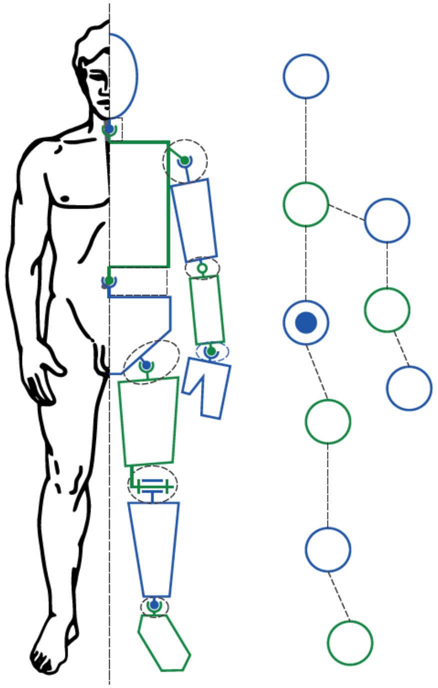
1. INTRODUCTION
➔ 2. METHODOLOGY & CONCEPT
3. RESULTS
4. DISCUSSION

Optical vs. Inertial
Musculoskeletal Pipeline

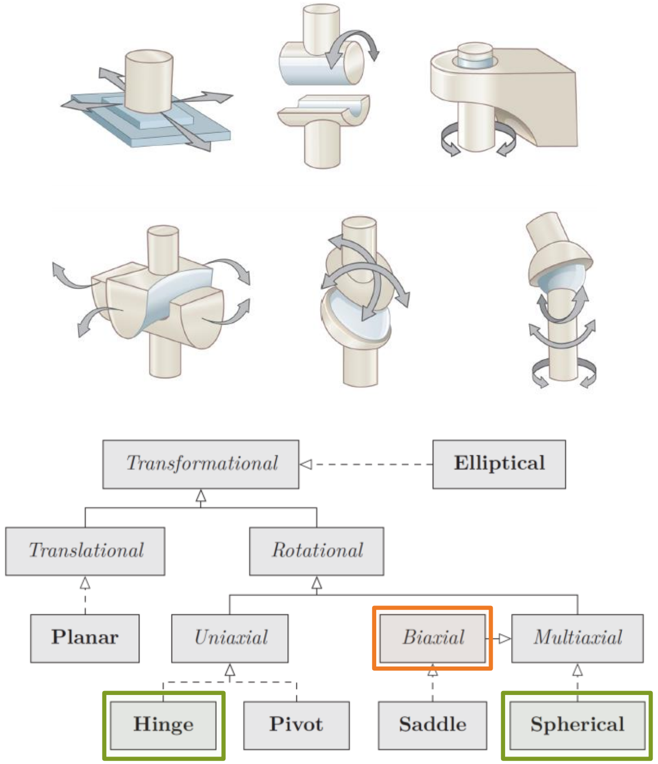


→ Optical Motion Capture + Inverse Kinematics

Kinematic Modeling



skeleton graph (topology)
of segments and joints



basic synovial joint types

spherical: no rotational restriction

hinge: one-axis restriction

➔ align frames of incident segments
+ restrict rotation using Euler angles

or:

penalize $r := a - aR$,

a: joint axis, R: rotation

biaxial: two-axes restriction,

➔ not trivial in general for non-perpendicular axes (cf. thesis)

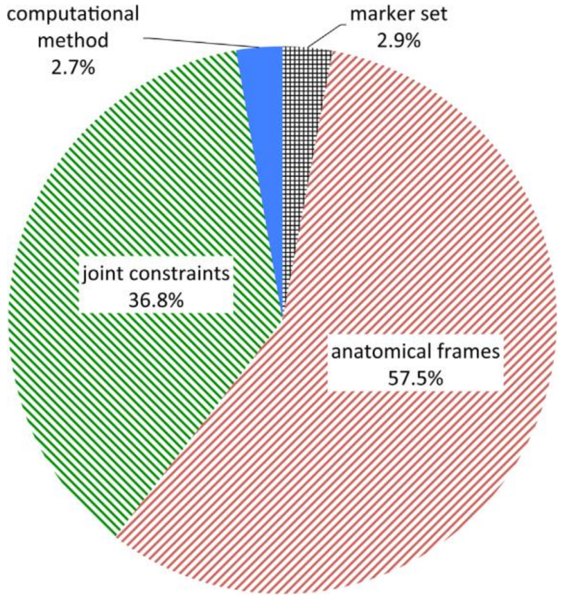
➔ later

common joint types in practice

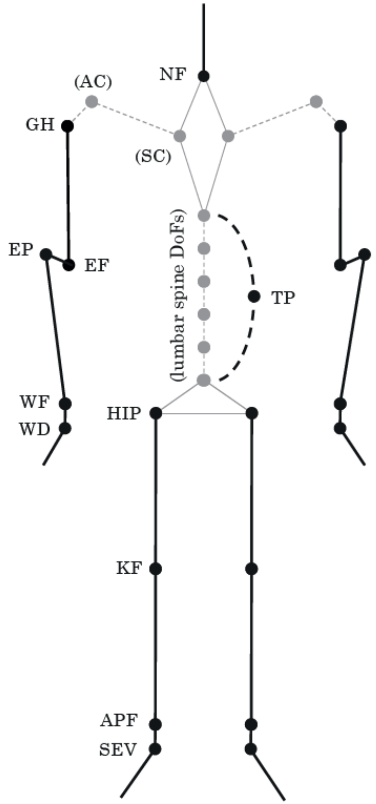
Anatomical Skeletons: AnyBody Full-Body Model

There are various anatomical skeleton models available!

but: kinematics is only comparable among the same underlying model



➔ **use the same, detailed, anatomically correct model for all stages of the pipeline (avoid the propagation of modeling errors through the entire pipeline!)**

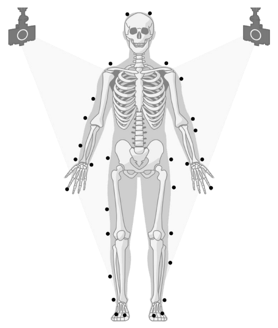


Abbrev.	DoFs	Description
NF	1×	Neck Flexion
TP	3×	Thorax Pelvis
(SCPV)	3×	Sacrum Pelvis
(L5SC)	3×	L5 Sacrum
(L4L5)	3×	L4 L5
(L3L4)	3×	L3 L4
(L2L3)	3×	L2 L3
(L1L2)	3×	L1 L2
(T12L1)	3×	T12 L1
GH	3×	Gleno Humeral
(SC)	3×	Sterno Calvicular
(AC)	3×	Acromio Clavicular
EF	1×	Elbow Flexion
EP	1×	Elbow Pronation
WF	1×	Wrist Flexion
WD	1×	Wrist Deviation
HIP	3×	Hip
KF	1×	Knee Flexion
APF	1×	Ankle Plantar Flexion
SEV	1×	Subtalar Eversion

"Joint kinematic calculation based on clinical direct kinematic versus inverse kinematic gait models."
Kainz et al. 2016, Journal of Biomechanics

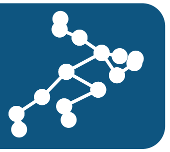
AnyBody skeleton model and DoFs

Classical Optical Marker-Based Body Tracking



Motion Capture

Inverse Kinematics



Model Scaling

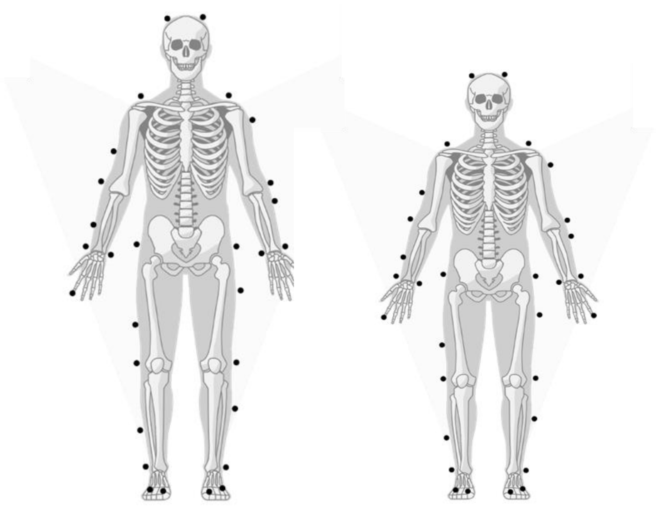
kinematic model

anthropometric measurements

regression models

Model Scaling

scaled kinematic model (joint-to-joint lengths)



Inverse Kinematics

$q(t)$: joint angle trajectories

$$\Gamma(q(t)) = 0$$ ← holonomic system

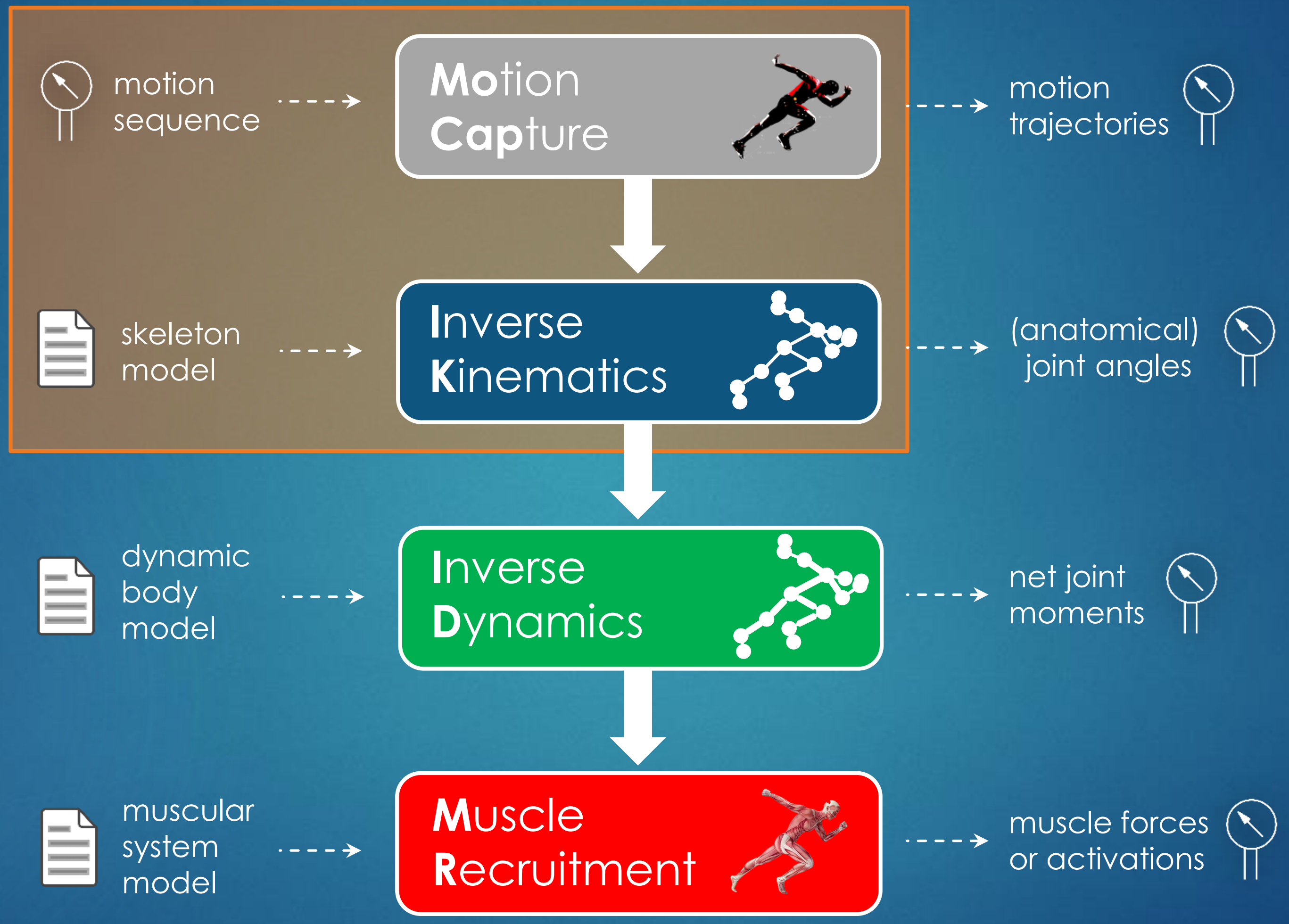
$$\Gamma(q(t)) = \begin{bmatrix} \Psi(q(t), P) \\ \Phi(q(t), P) \end{bmatrix}$$

← marker & motion constraints, dep. on model scaling

← skeleton joint constraints, dep. on model scaling

➔ precise kinematics ➔ optical marker-based body tracking = "golden standard"

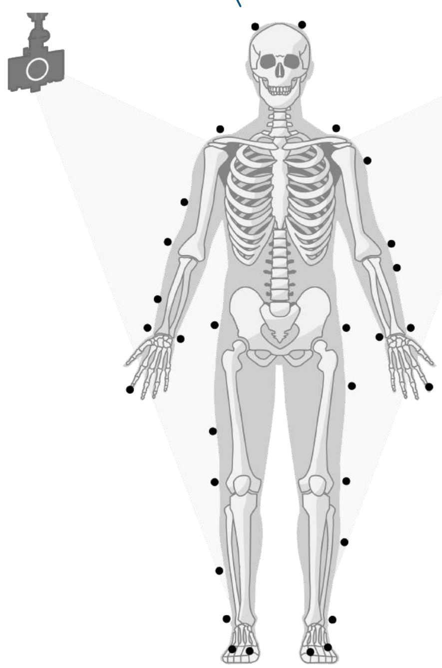
Optical vs. Inertial
Musculoskeletal Pipeline



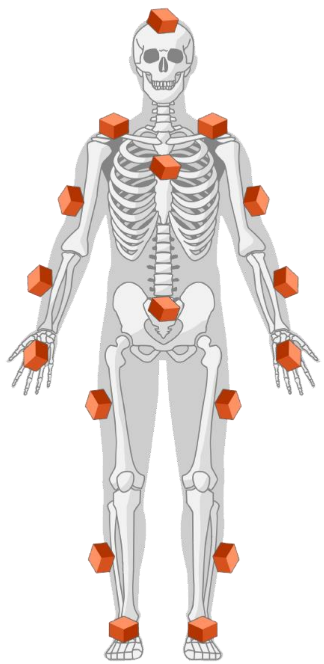
→ Inertial Motion Capture + Inverse Kinematics

Comparison of Optical vs. Inertial Body Tracking

optical markers



Inertial Measurement Units (IMUs)



optical: markers at bony landmarks

➔ global body reference frame

➔ marker tracking optimization

➔ marker placement errors (soft tissue artifacts)

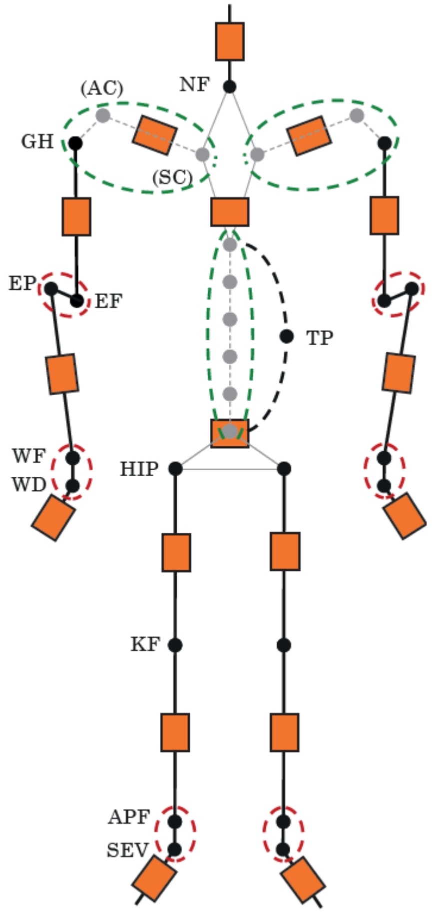
inertial: sensors at segments

➔ local inertial sensor frames

➔ inertial multi-body tracking problem

➔ **I2S calibration errors**

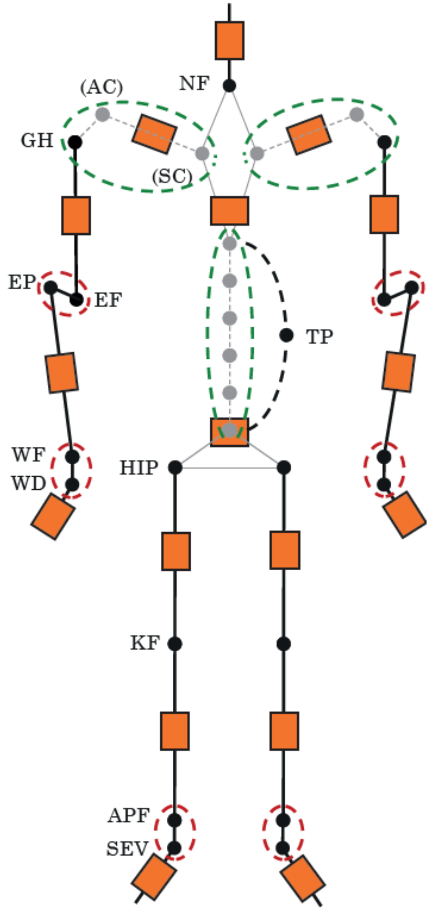
one-to-one sensor-to-segment mapping is not feasible!



Abbrev.	DoFs	Description
NF	1×	Neck Flexion
TP	3×	Thorax Pelvis
(SCPV)	3×	Sacrum Pelvis
(L5SC)	3×	L5 Sacrum
(L4L5)	3×	L4 L5
(L3L4)	3×	L3 L4
(L2L3)	3×	L2 L3
(L1L2)	3×	L1 L2
(T12L1)	3×	T12 L1
GH	3×	Gleno Humeral
(SC)	3×	Sterno Calvicular
(AC)	3×	Acromio Clavicular
EF	1×	Elbow Flexion
EP	1×	Elbow Pronation
WF	1×	Wrist Flexion
WD	1×	Wrist Deviation
HIP	3×	Hip
KF	1×	Knee Flexion
APF	1×	Ankle Plantar Flexion
SEV	1×	Subtalar Eversion

➔ **detailed skeleton model is not directly IMU-trackable**

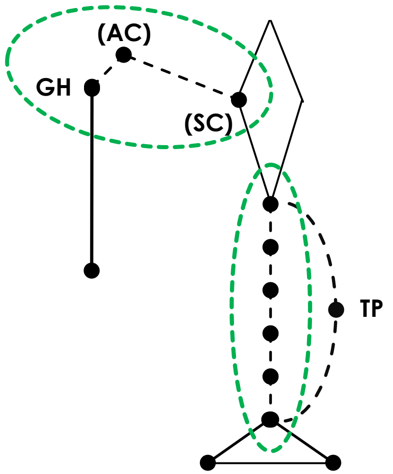
IMU-Trackable Anatomical Skeleton (Shoulder and Spine Rhythms)



green: too many DoFs compared to IMUs

AnyBody skeleton implements shoulder and spine kinematics as a functions of **only one joint**

➔ **„shoulder & spine rhythm"** extracted from AnyBody code and implemented



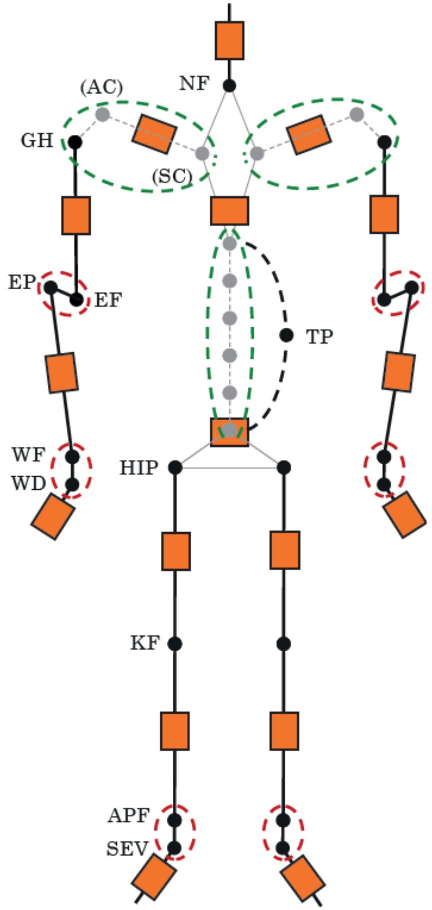
shoulder rhythm: **[AC, SC] = $f_{shoulder}$ (GH)**

spine rhythm: **[SacrumPelvis, L5Sacrum, L4L5, L3L4, L3L2, L1L2, T12L1] = f_{spine} (TP)**

***f* (.)** *are linear combinations*

"*Shoulder Rhythm Report*" and "*Spine Rhythm*" Technical Reports, AnyBody Technology A/S

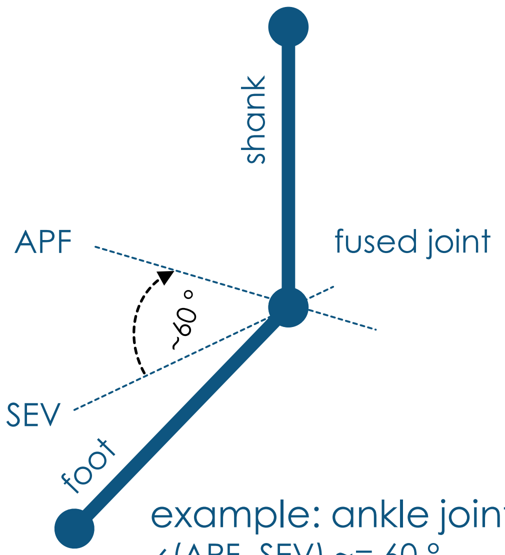
IMU-Trackable Anatomical Skeleton (Arbitrary two-Axes Joint Constraint)



red: DoFs trackable in principle, but redundant in-between segment

redundant in-between segments ➔ shrink segments to zero length (fuse joints)

➔ simple, if joint axes are perpendicular:
obtain relative orientaton between incident IMUs and decompose it (Euler angles)



example: ankle joint
∠(APF, SEV) ~= 60 °

➔ is there a rotation decomposition about „arbitrary axes“?

two-axes decomposition:

with R and two non-collinear axes a_1 and a_2,
R is decomposable about a_1, a_2 iff:

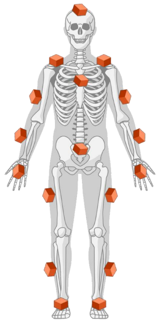
$$a_1^T a_2 = a_1^T R a_2$$

$r := |a_1^T a_2 - a_1^T R a_2|$ is a measure of the remaining "non-decomposability" of R

➔ use $\min r$ as a joint constraint

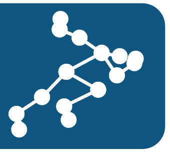
"*On Coordinate-Free Rotation Decomposition: Euler Angles about Arbitrary Axes.*" Piovan, Bullo 2012

Inertial Inverse Kinematics



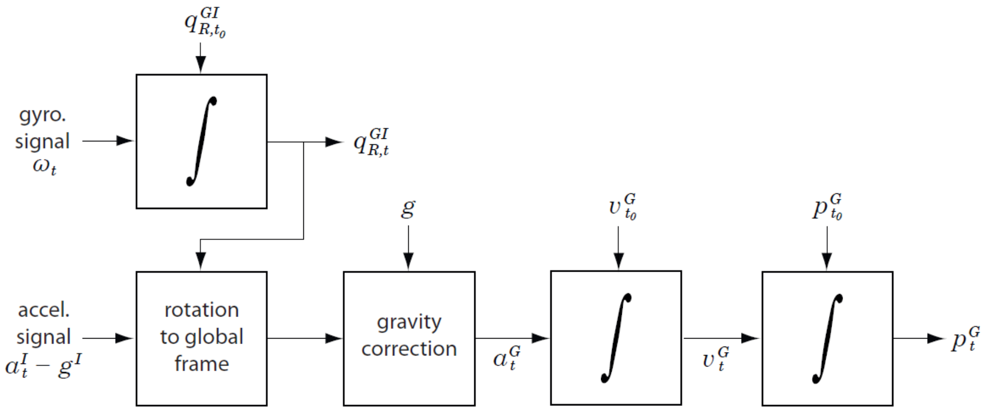
Motion
Capture

Inverse
Kinematics



Complementary Inertial Sensor Fusion / Strapdown Integration:

accelerometer + gyroscope (+ magnetometer)



Multi Sensor Fusion: Extended Kalman Filter (EKF)

Coupled Fusion Approach: QuatTracker

EKF

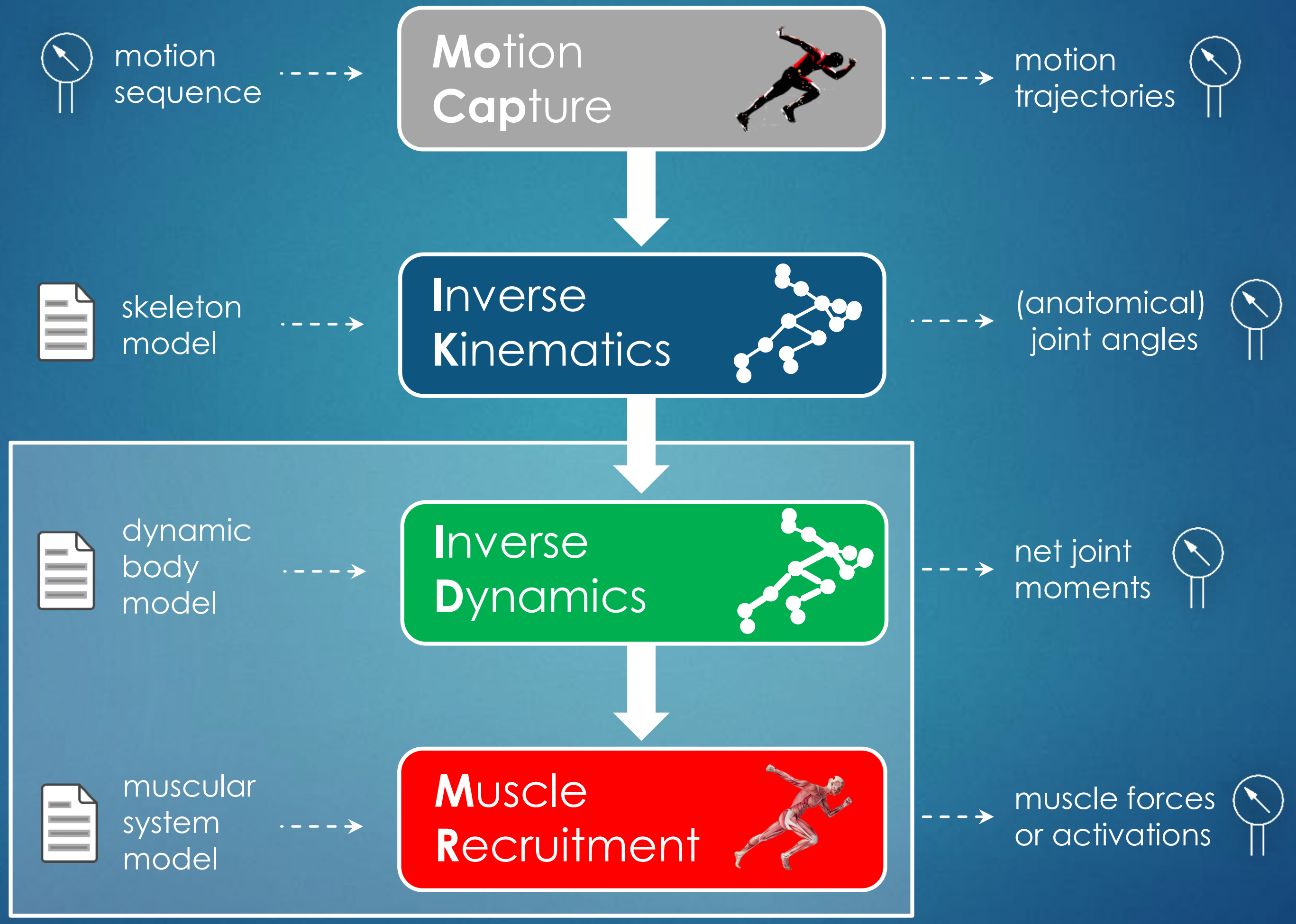
Strapdown

QuatTracker

tackable skeleton model
(+ two-axes joint constraints
+ shoulder & spine rhythm)

"*On inertial body tracking in the presence of model calibration errors*", Miezal, Taetz, Bleser, Sensors, vol. 16, no. 7, p. 1132, 2016

Optical vs. **Inertial**
Musculoskeletal Pipeline



→ Inverse Dynamics + Static Optimization

Dynamic Modeling

scaled kinematic model

body mass distributions

Model Scaling

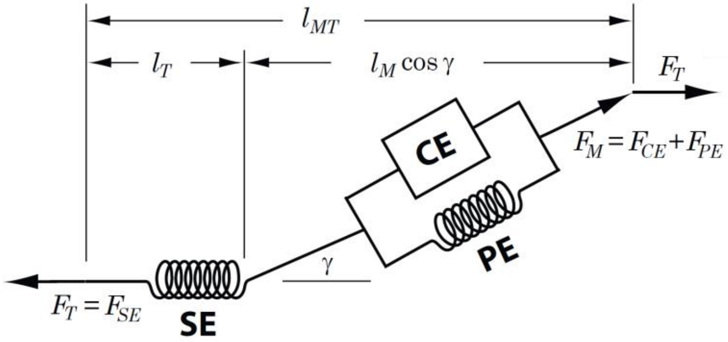
body fat percentage (BMI)

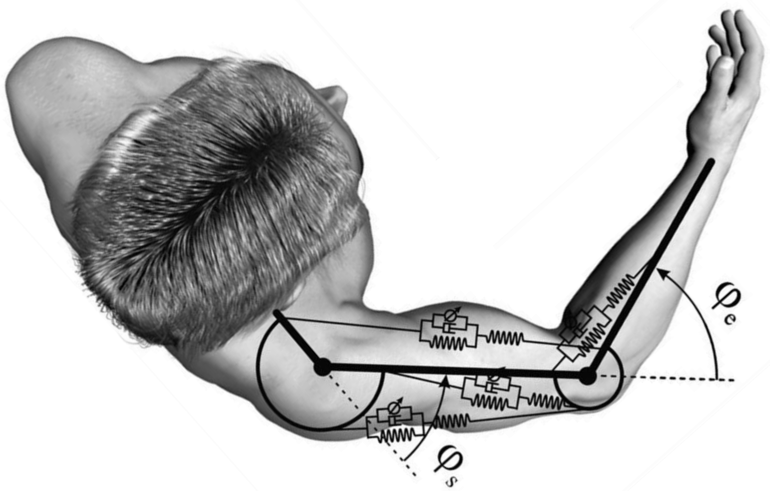
scaled dynamic model

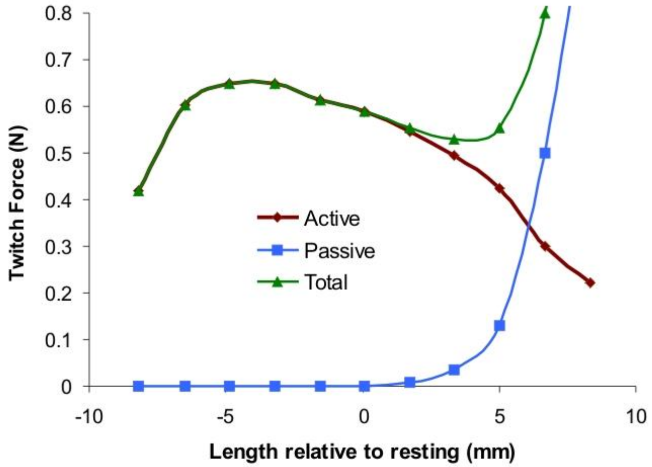
dynamic body model (masses and inertia)

+

muscular system model (muscle geometry)

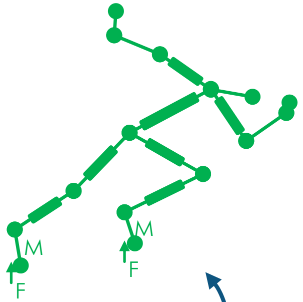






basic hill type muscle model (musculotendon unit)

Inverse Dynamics



mapping?

Inverse
Dynamics

Muscle
Recruitment

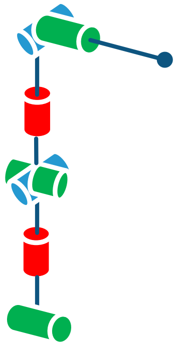
Equations of Motion

$$\begin{bmatrix} m_i I_3 & 0 \\ 0 & \Theta_i \end{bmatrix} \begin{bmatrix} \dot{v}_i \\ \dot{\omega}_i \end{bmatrix} + \begin{bmatrix} 0 \\ \omega_i \times \Theta_i \, \omega_i \end{bmatrix} = \begin{bmatrix} f_i \\ \tau_i \end{bmatrix}$$

net forces
and torques
➔ muscle forces?

Muscular Redundancy Problem

- *"How to distribute net joint moments over all involved muscles?"*



example: arm without hand

- 7 DoFs
- but > 20 skeletal muscles

➔ joints are over-actuated ⇔ muscle system is underdetermined

➔ muscle recruitment solution requires additional constraints

Muscle Recruitment Optimization

Idea of Static Optimization

in each time step, choose a solution that minimizes the total muscle stress while fulfilling the force balance prescribed by the equations of motion:

$$\hat{f}^{(\mathbb{M})} = \underset{f^{(\mathbb{M})}}{\arg\min} \left[G\left(f^{(\mathbb{M})}\right) \right]$$

$$\text{s.t. } C\,f = \tau_q$$

Muscle Recruitment Criterion

$$G\left(f^{(\mathbb{M})}\right) = \sum_{M \in \mathbb{M}} \left(\frac{f_{M_i}}{F_{M_i,0}} \right)^p$$

cf. polynomial norms

$p = 1$: linear recruitment: allocate more work to stronger muscles

➔ minimum number of muscles for equilibrium

$p = 2$: quadratic recruitment: penalize large single force terms

➔ emphasize load sharing, i.e. muscle synergy effects

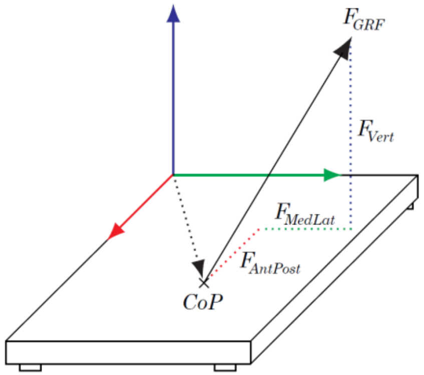
$p = \infty$: min-max recruitment: balance loads among muscles as best as possible

➔ minimum muscle fatigue or maximum synergism criterion

Inverse Dynamics

Muscle **R**ecruitment

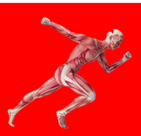
Ground Reaction Forces



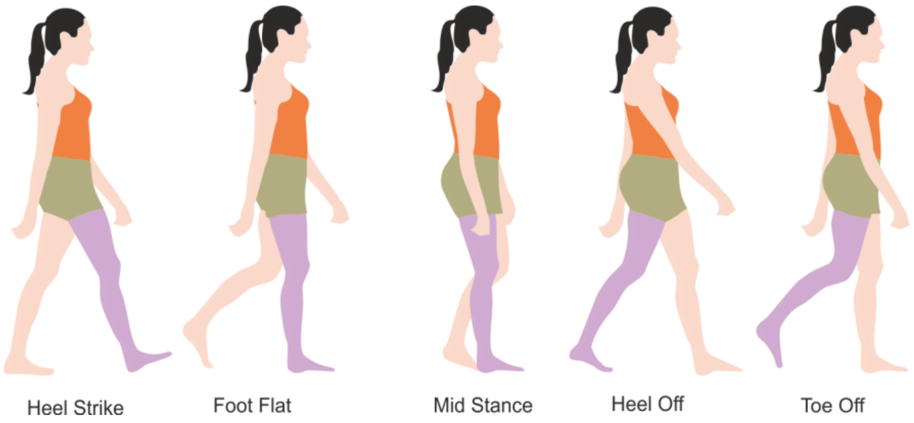
$$\begin{bmatrix} m_i I_3 & 0 \\ 0 & \Theta_i \end{bmatrix}\begin{bmatrix}\dot{v}_i \\ \dot{\omega}_i\end{bmatrix} + \begin{bmatrix} 0 \\ \omega_i \times \Theta_i \, \omega_i\end{bmatrix} = \begin{bmatrix} f_i \\ \tau_i\end{bmatrix}$$

Inverse Dynamics

Static Optimization



Why and how to measure GRFs?



➔ GRFs indetermined during double feet contact phases (closed kinematic chain)

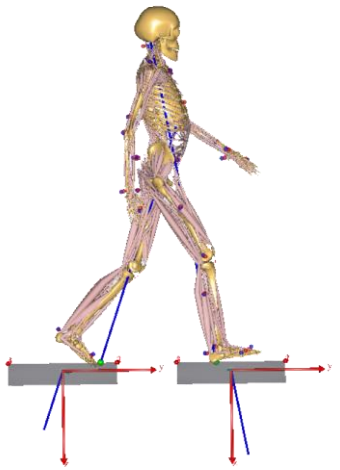
Limitations of GRF Measurements

- possible dynamic inconsistency of equations of motions for measured GRFs
- **inconvenient and inflexible force plate devices**

➔ **GRF prediction** using additional assumptions / constraints ... ?

- optimization based approach [Audu et al. 2003]
- smooth transitions functions, interpolated from empirical measurements [Ren et al. 2008]
- artificial neural networks [Choi et al. 2013]
- zero-moment point (robotics)
- or ... ?

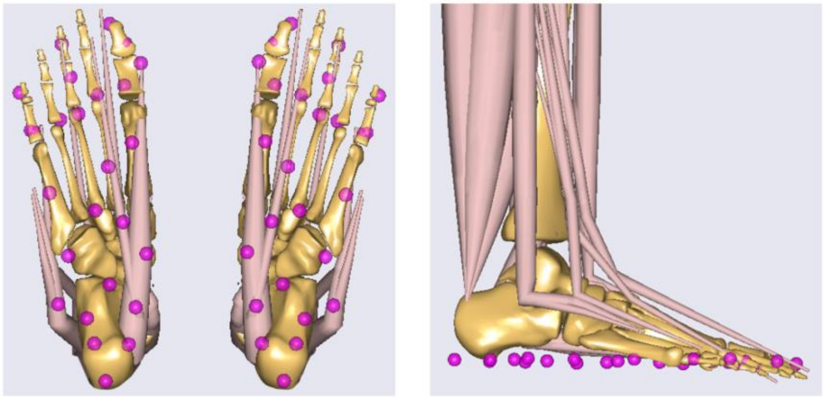
Ground Reaction Force Prediction



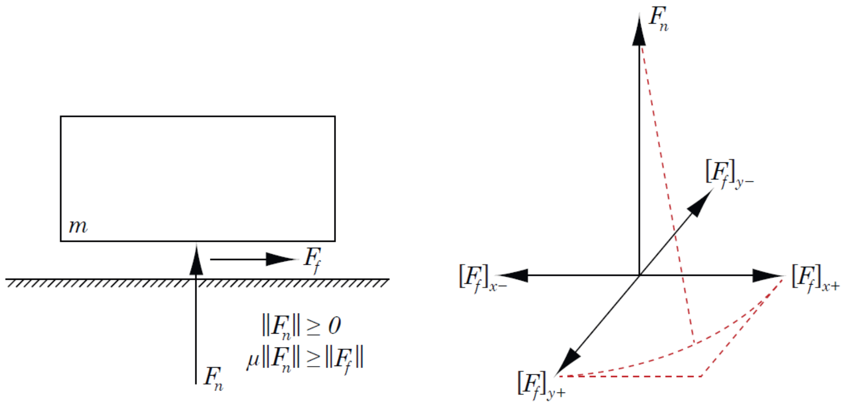
Inverse Dynamics

Static Optimization

Idea: Artificial Muscle-Like Actuators at Foot Ground Contact Points



- 25 **ground contact points** per foot
- 5 **muscle-like actuators** per point
- **ground contact conditions** for each point:
    - zero-velocity condition
    - height threshold



$$||F_f|| \leq \sqrt{[F_f]_x^2 + [F_f]_y^2} \leq \mu\,||F_n||$$

➔ integration into Static Optimization

$$\hat{f}^{(\mathbb{M})} = \underset{f^{(\mathbb{M})}}{\arg\min} \left[G\left(f^{(\mathbb{M})}\right) \right]$$

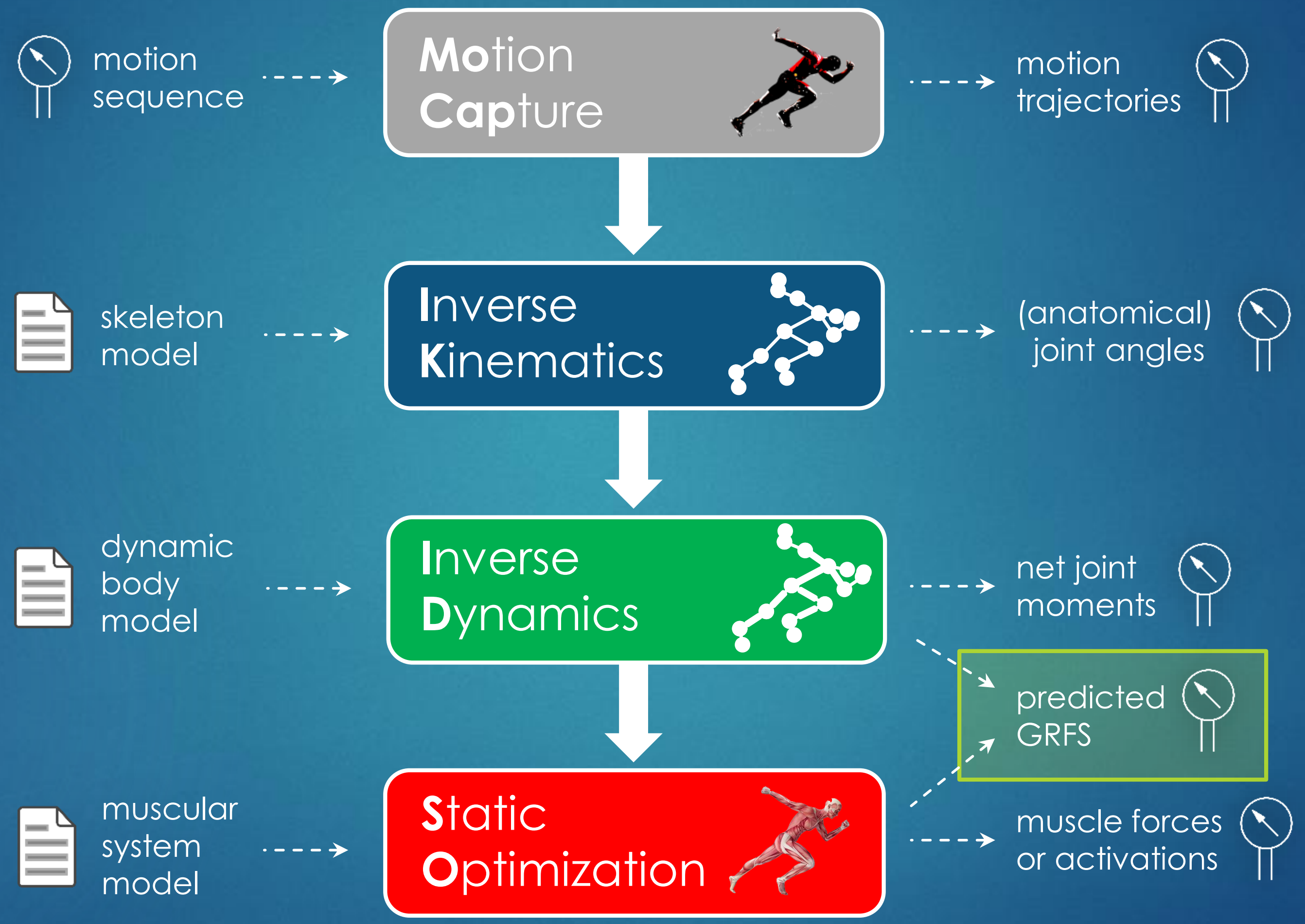
$$\text{s.t. } C\,f = \tau_q$$

"*Prediction of ground reaction forces and moments during various activities of daily living.*" Fluit et al. 2014, Journal of Biomechanics

Agenda

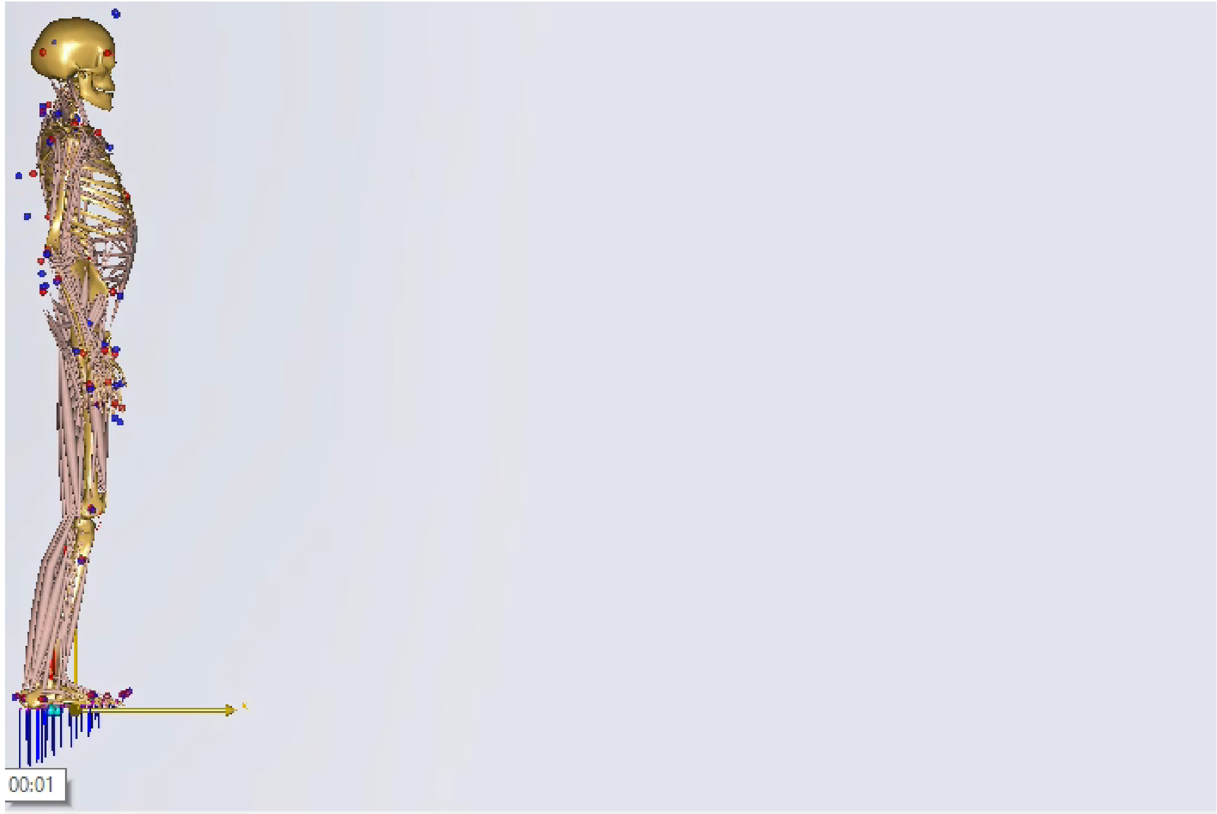
1. INTRODUCTION 2. METHODOLOGY & CONCEPT ➔ 3. RESULTS 4. DISCUSSION

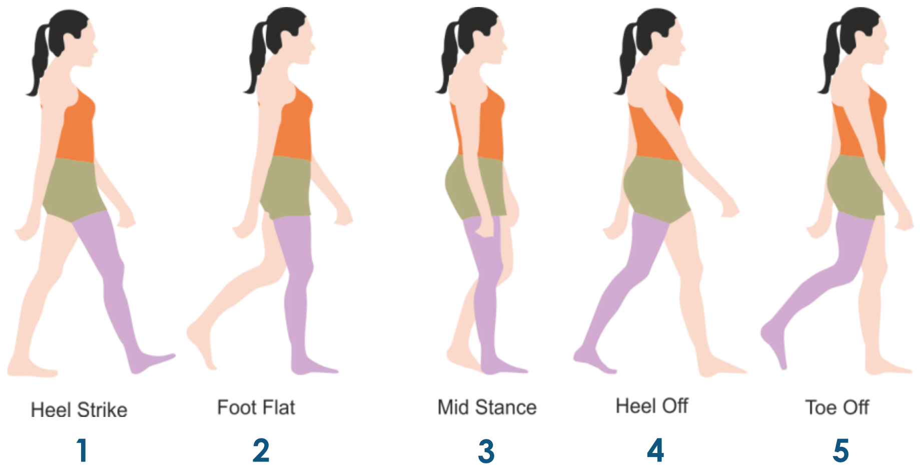
Optical vs. Inertial
Musculoskeletal Pipeline



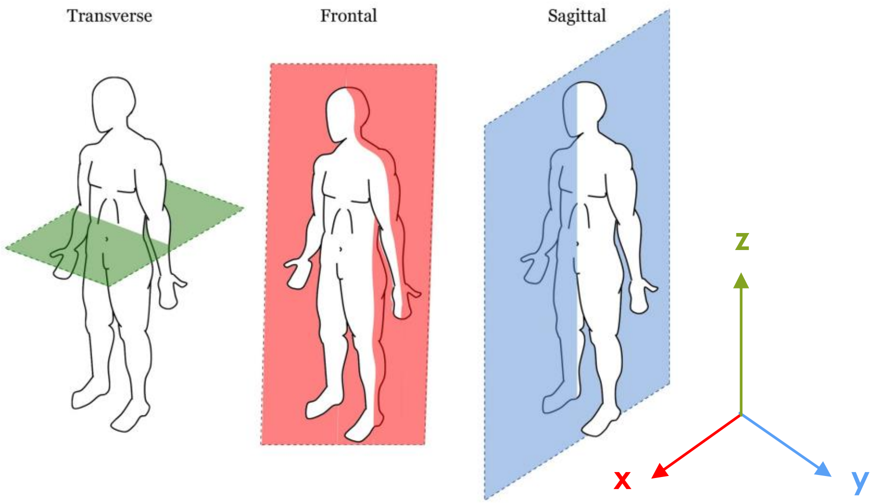
➔ GRF Prediction: Validation & Sensitivity Analysis

GRF Prediction Validation





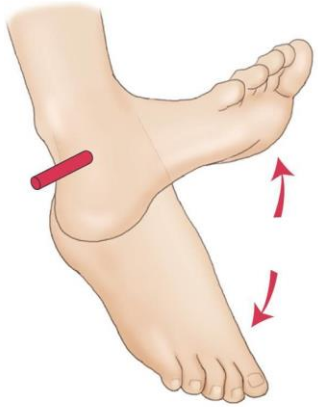
gait cycle with gait events



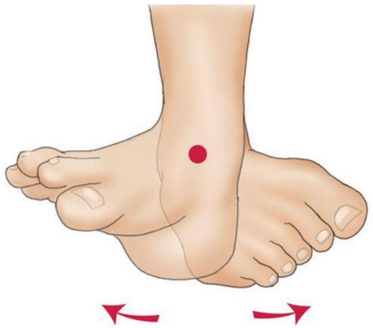
$N = 3$	F_x (ant-pos)	F_y (med-lat)	F_z (ver)	M_x (fro)	M_y (sag)	M_z (tra)
$\bar{r}(*, \rightarrow)$	0.95	0.74	0.99	0.73	0.97	0.78
$\bar{e}_R(*, \rightarrow)$	0.07	0.27	0.05	0.22	0.06	0.18

➔ results in accordance with [Fluit et al. 2014] and [Skals 2016]

GRF Prediction Sensitivity Analysis (Ground Contact Conditions)

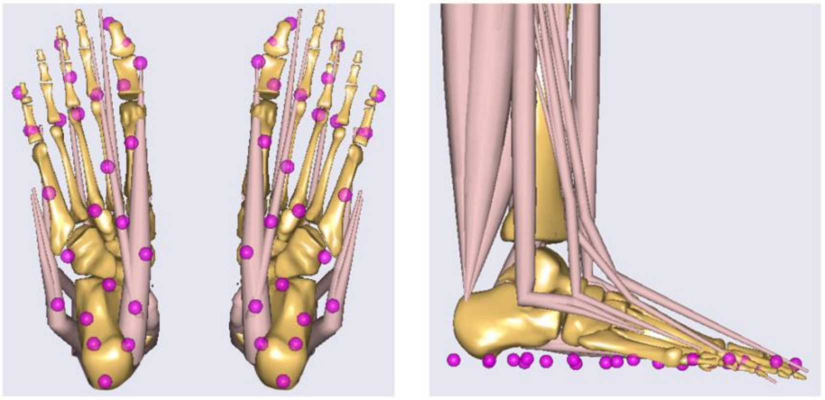


Ankle Plantar Flexion (APF)



Subtalar Eversion (SEV)

error sampling range of [±12 ° x ±12 °]



APF + SEV ➔ foot posture during touchdown

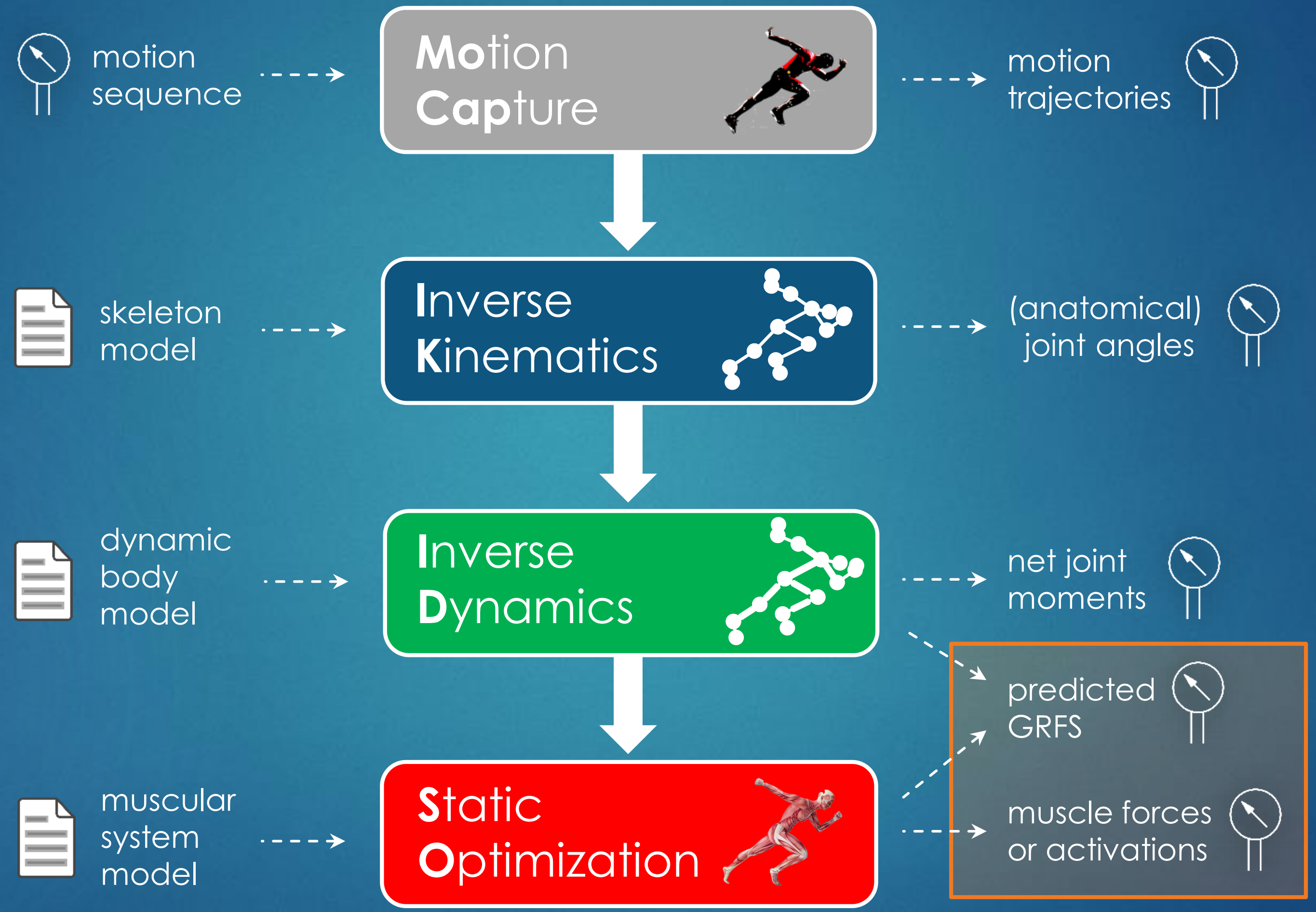
	F_x (ant-pos)	F_y (med-lat)	F_z (ver)	M_x (fro)	M_y (sag)	M_z (tra)
$\bar{r}(*, \rightarrow)$	0.99	0.99	0.99	0.86	0.99	0.98
$\tilde{r}(*, \rightarrow)$	0.99	0.99	0.99	0.92	0.99	0.99
$\check{r}(*, \rightarrow)$	0.98	0.97	0.99	0.18	0.98	0.78
$s_r(*, \rightarrow)$	0.0018	0.0054	0.0002	0.166	0.0032	0.0334
$\bar{e}_R(*, \rightarrow)$	0.0066	0.0352	0.0054	0.1343	0.0179	0.0653
$\tilde{e}_R(*, \rightarrow)$	0.0051	0.0341	0.0050	0.1383	0.0159	0.0543
$\hat{e}_R(*, \rightarrow)$	0.0432	0.1036	0.0215	0.2435	0.0794	0.2456
$s_{e_R}(*, \rightarrow)$	0.0059	0.0203	0.0031	0.0616	0.0134	0.0488

➔ GRF prediction is robust against foot posture disturbances

$r(\downarrow, \rightarrow)$	$\delta[\mathrm{CoP}]_{\bar{x}}$ (ant-pos)	$\delta[\mathrm{CoP}]_{\bar{y}}$ (med-lat)
$\|\|\theta_{APF}\|\|$	0.82	0.27
$\|\|\theta_{SEV}\|\|$	0.23	0.69

➔ tradeoff between prediction robustness and CoP precision

Optical vs. Inertial
Musculoskeletal Pipeline



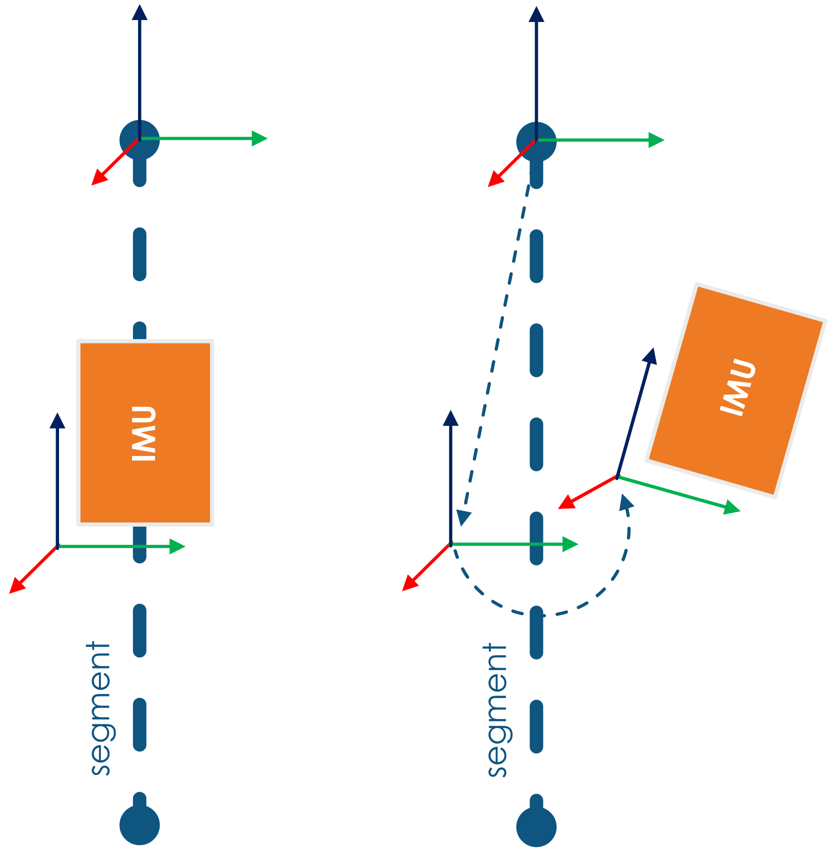
→ I2S Calibration Error Simulation

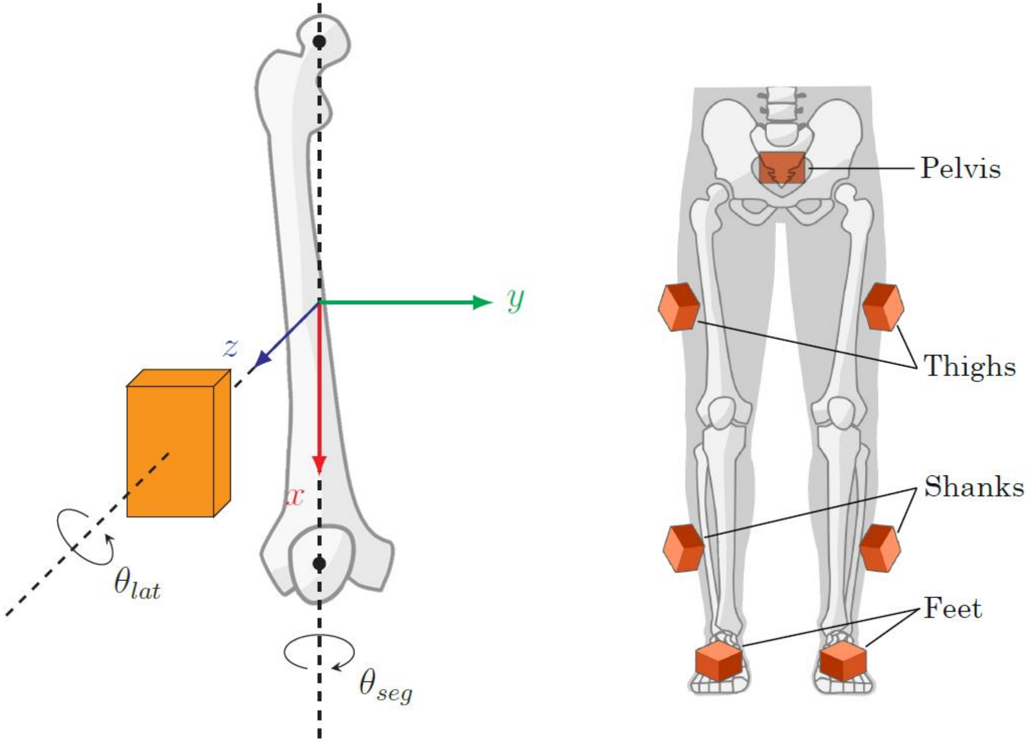
I2S Calibration Error Simulation

Effects of simulated I2S calibration errors on predicted GRFs

➔ how accurate are biomechanical analysis results w.r.t. I2S errors?

➔ what calibration precision is required?



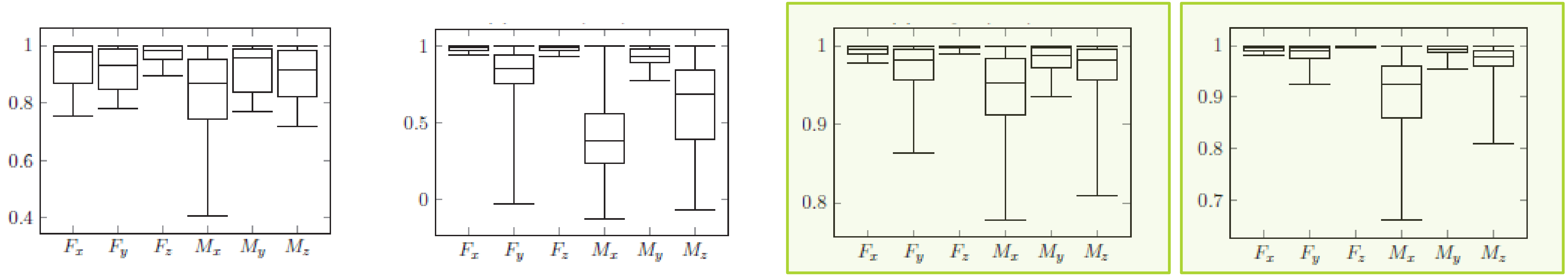


simulated errors around lateral and segmental axes

I2S Calibration Errors: Effects on Predicted GRFs per Segment

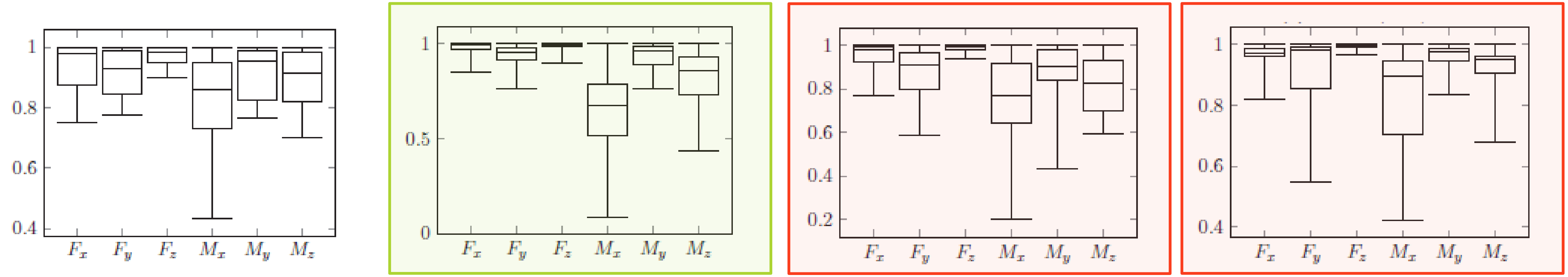
correlations of disturbed vs. true predicted GRFs for **anterior** initial calibration, error sampling **range of [±12 ° x ±12 °]**

anterior initial calibration



Pelvis **Thigh** **Shank** **Foot**

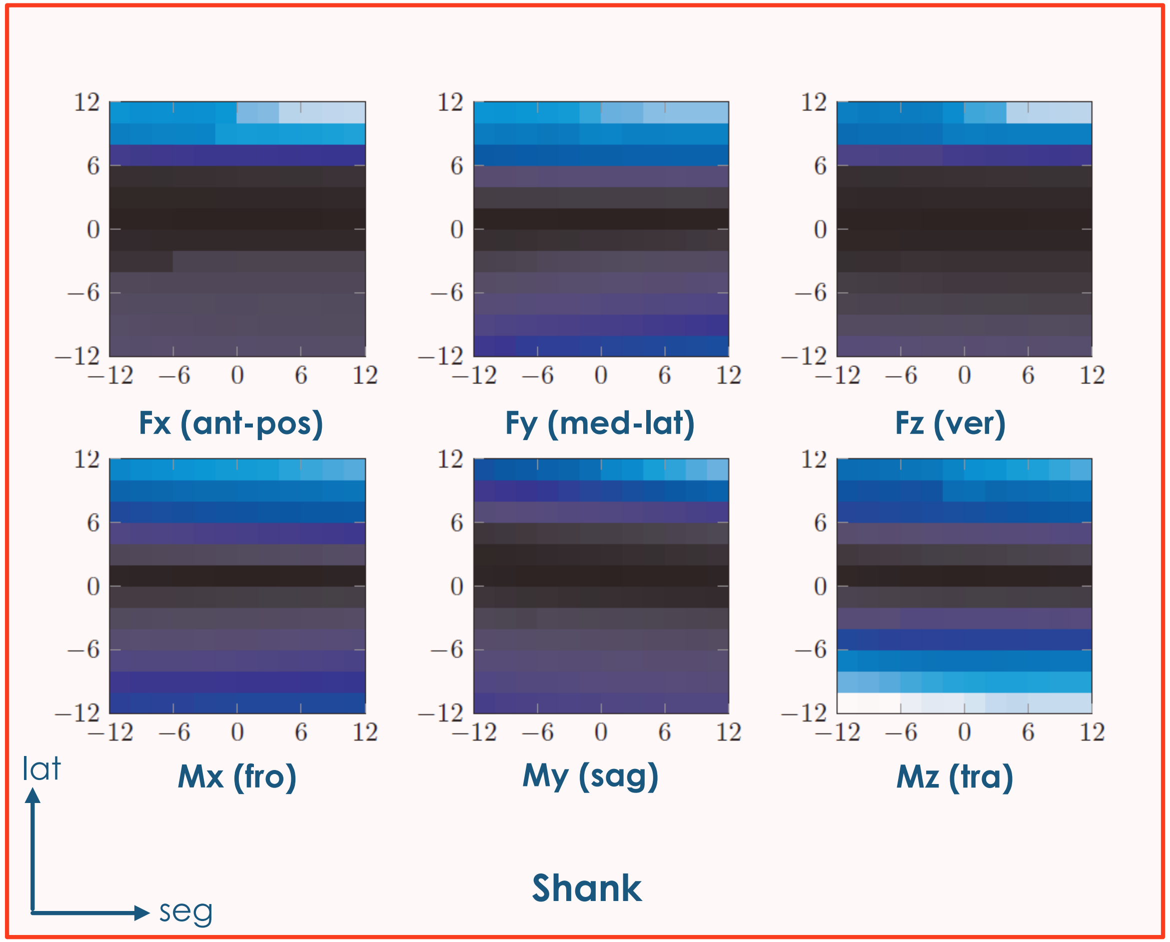
lateral initial calibration

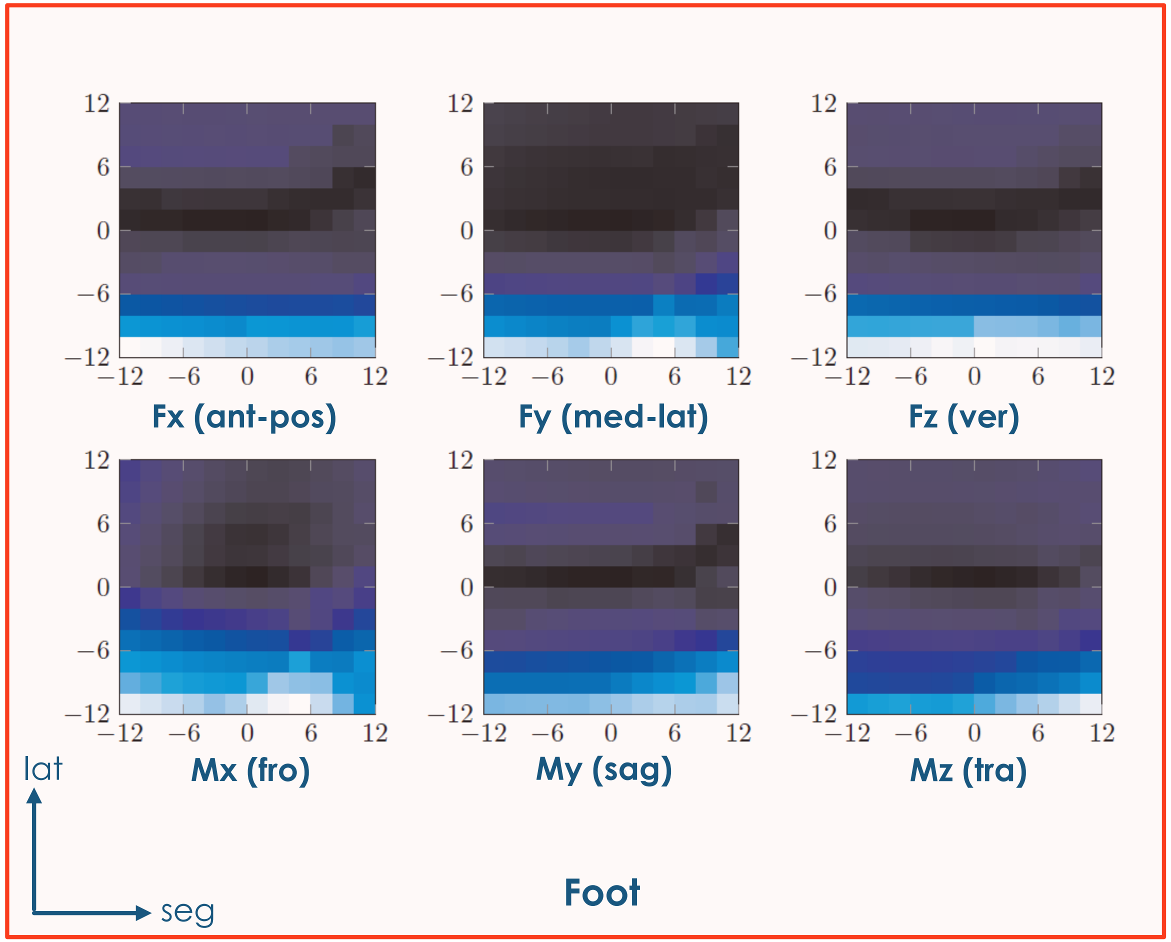


correlations of disturbed vs. true predicted GRFs for **lateral** initial calibration, error sampling **range of [±12 ° x ±12 °]**

I2S Calibration Errors: Effects on Predicted GRFs per Segment

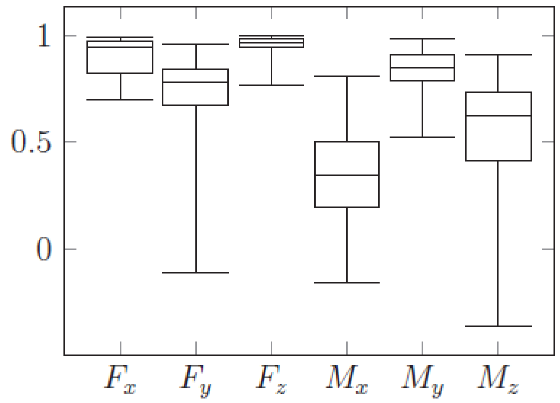
qualitattive correlation distributions for **lateral** initial calibration, error sampling **range of [±12 ° x ±12 °]**



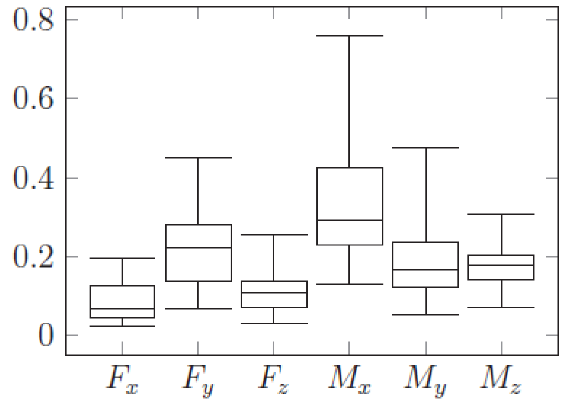


I2S Calibration Errors: Effects on Predicted GRFs in Typical Scenario

monte carlo simulation of I2S errors for **all lower body segments** in an error **range of [±12 ° x ±12 °]**

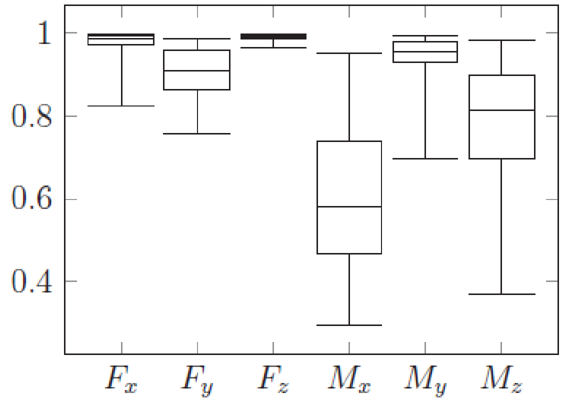


(a) Boxplots of $r(*, \rightarrow)$.

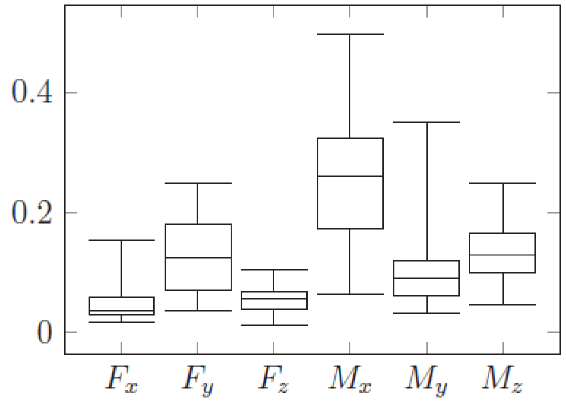


(b) Boxplots of $e_R(*, \rightarrow)$.

monte carlo simulation of I2S errors for **all lower body segments** in a **reduced range of [±6 ° x ±6 °]**



(a) Boxplots of $r(*, \rightarrow)$.



(b) Boxplots of $e_R(*, \rightarrow)$.

Agenda

1. INTRODUCTION 2. METHODOLOGY & CONCEPT 3. RESULTS → 4. DISCUSSION

Conclusion

- ✓ musculoskeletal analysis pipeline for both optical and inertial body tracking
- ✓ anatomical IMU-trackable skeleton, consistently used in the entire pipeline
  - ➔ avoid (unknown) modeling errors
- ✓ integration of a universal GRF prediction approach
  - ➔ "universal" = no training / empirical data required, only kinematics

- ✓ sensitivity of GRF Prediction (ground contact conditions)
  - ➔ very robust for moderate foot posture disturbances
- ✓ systematic I2S calibration error simulation
  - ➔ robustness dependent on initial IMU configuration per segment
  - ➔ generally: much higher impact of errors on lateral axis
  - ➔ acceptable GRF prediction errors for proper initial configuration
- ✓ typical I2S error scenario for all lower body segments
  - ➔ errors at segments accumulate → moderate to high GRF prediction deviations
  - ➔ acceptable errors for reduced error range of [±6 ° x ±6 °]

Future Work

- quantitatively proof the advantages of using the same model for the entire pipeline
  ➔ compare with kinematics + dynamics obtained with different models

- repeat studies with real data
  ➔ use real inertial reference trajectories
  ➔ use force plate measurements

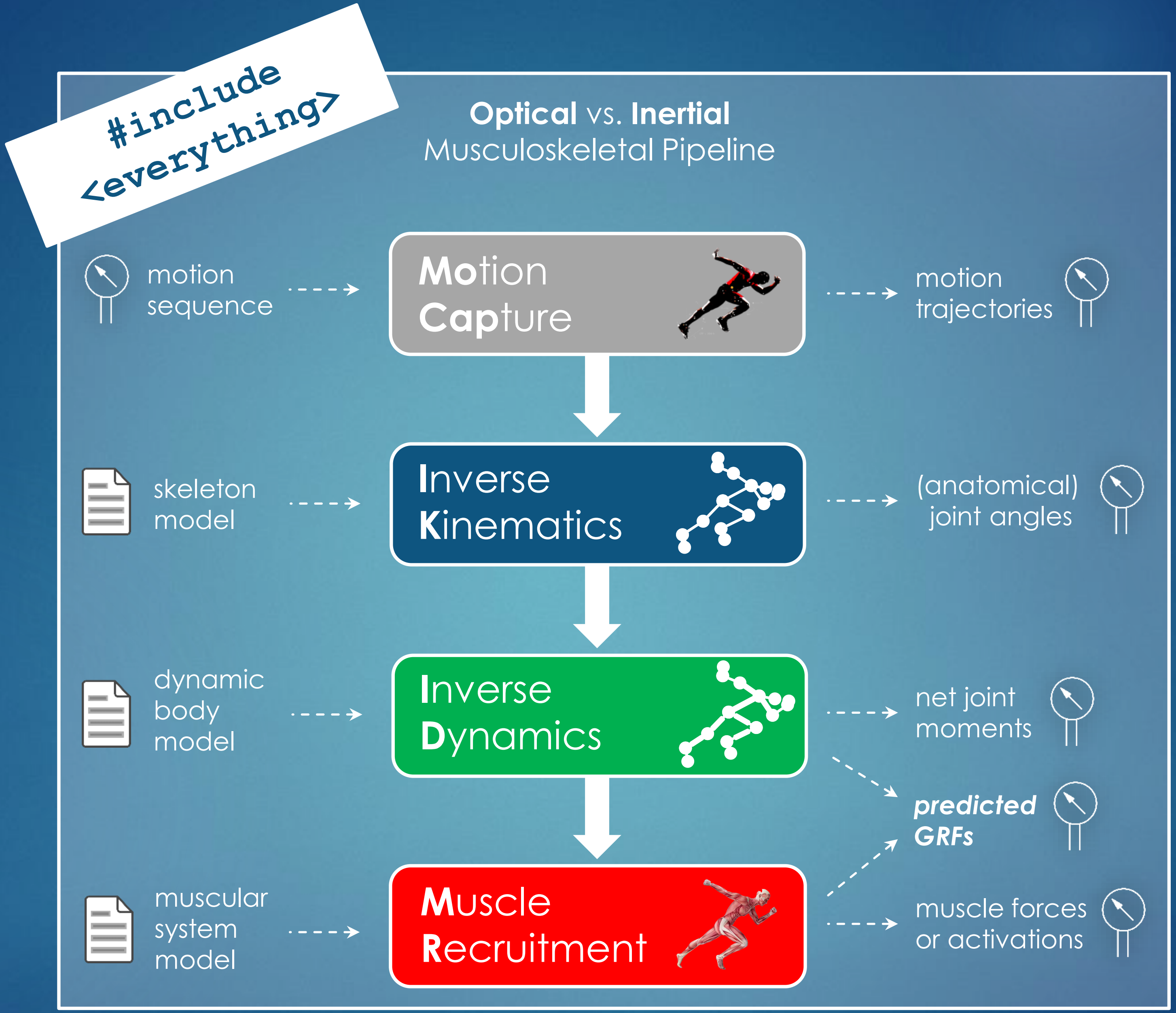
- what about pathological gaits / motion sequences?
  ➔ GRF prediction valid?

- lower body ➔ full-body I2S error simulation

- (investigate muscle activation estimations in more detail)
- (can we make the inverse dynamics computationally more efficient?)
- (integrate and compare different biomechanical frameworks and models, e.g. OpenSim vs. AnyBody)

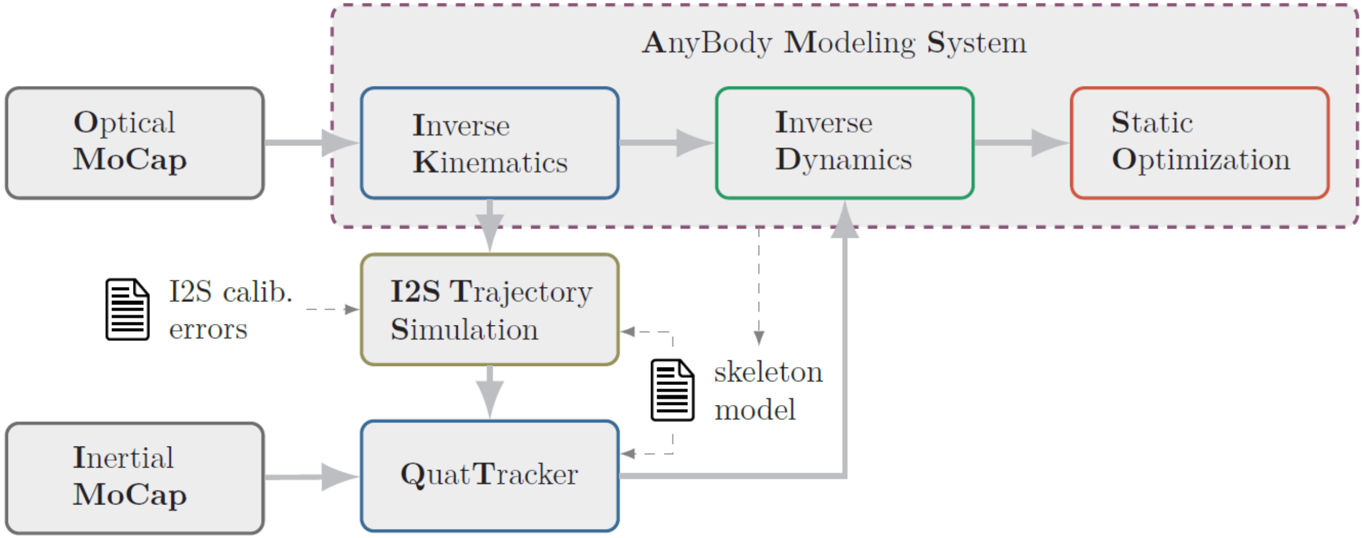
Questions?

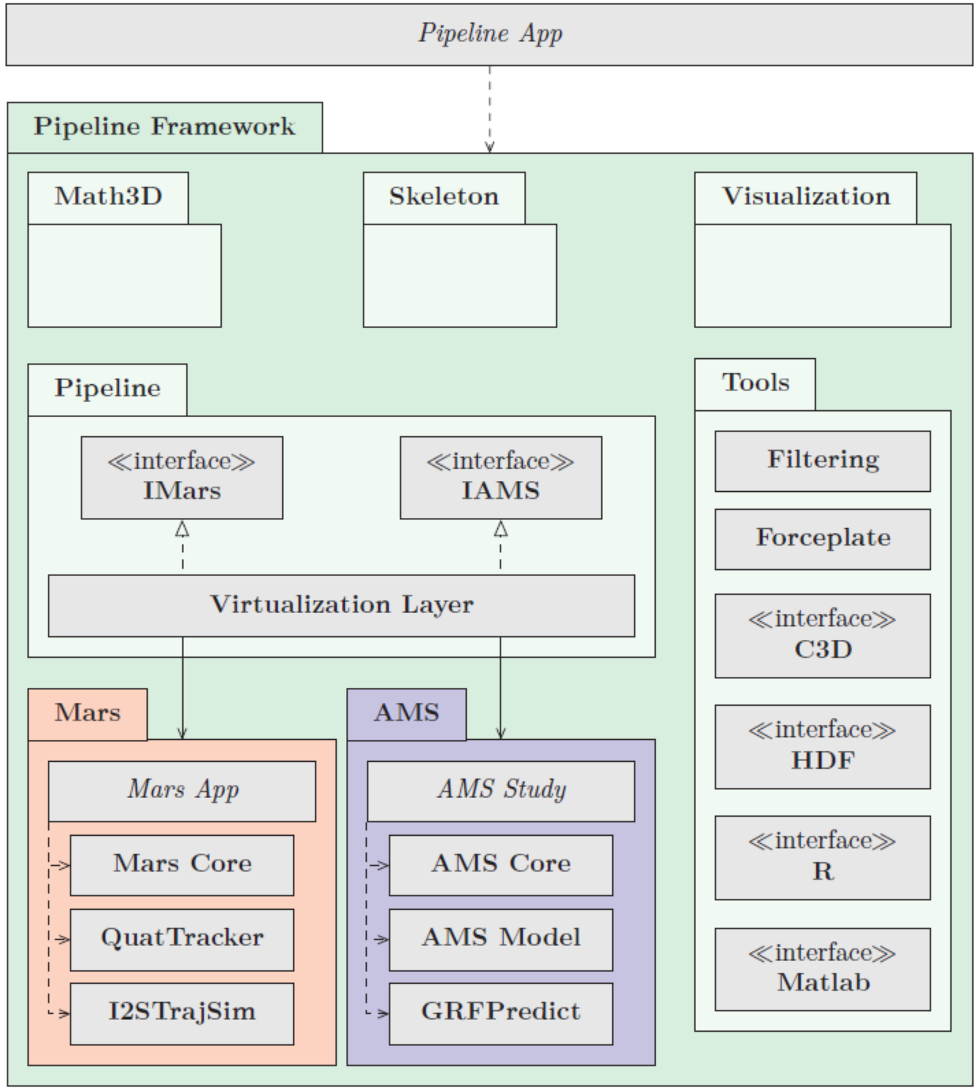
Thank you for your attention!



→ Overall System Design & Implementation

Overall Design & Implementation





I2S Trajectory Simulation

- data differentiation
- realistic inertial sensor error + noise model

Shoulder Rhythm Implementation

```
public static Dictionary<string, EulerAngles> ShoulderRhythmDOFs(EulerAngles glenohumeralLeftDOFs, EulerAngles glenohumeralRightDOFs)
{
    glenohumeralLeftDOFs = glenohumeralLeftDOFs.ToEulerAngles("XZY");
    glenohumeralRightDOFs = glenohumeralRightDOFs.ToEulerAngles("XZY");

    // Functions from shoulder rhythm definition in file /Body/AAUHuman/Arm/JntSR.any
    var sternoClavicularLeftDOFs = new EulerAngles(
        0.422 * glenohumeralLeftDOFs.X - 0.423,
        -0.242 * glenohumeralLeftDOFs.X + 0.12 * glenohumeralLeftDOFs.Z + 0.851 * -0.401 - 4.983.ToRad() + 10d.ToRad(),
        0.123 * glenohumeralLeftDOFs.X - 0.046 * glenohumeralLeftDOFs.Z + 0.493 * 0.201 + 3.917.ToRad() - 6d.ToRad()
    , "YZX");
    var sternoClavicularRightDOFs = new EulerAngles(
        0.422 * glenohumeralRightDOFs.X - 0.423,
        -0.242 * glenohumeralRightDOFs.X + 0.12 * glenohumeralRightDOFs.Z + 0.851 * -0.401 - 4.983.ToRad() + 10d.ToRad(),
        0.123 * glenohumeralRightDOFs.X - 0.046 * glenohumeralRightDOFs.Z + 0.493 * 0.201 + 3.917.ToRad() - 6d.ToRad()
    , "YZX");
    var sign = +1d;
    var scapulaThoraxLeftDOFs = new EulerAngles(
        0d,
        -0.049 * sign * glenohumeralLeftDOFs.X + 0.14 * sign * glenohumeralLeftDOFs.Z + sign * -1.203.ToRad() + 0.901 * 0.33 + 10d.ToRad(),
        0.396 * sign * glenohumeralLeftDOFs.X - 0.079 * sign * glenohumeralLeftDOFs.Z  + sign * 3.095.ToRad() + 0.414 * 0.307 - 10d.ToRad()
    , "YZX");
    sign = -1d;
    var scapulaThoraxRightDOFs = new EulerAngles(
        0d,
        -0.049 * sign * glenohumeralLeftDOFs.X + 0.14 * sign * glenohumeralLeftDOFs.Z + sign * -1.203.ToRad() + 0.901 * 0.33 + 10d.ToRad(),
        0.396 * sign * glenohumeralLeftDOFs.X - 0.079 * sign * glenohumeralLeftDOFs.Z + sign * 3.095.ToRad() + 0.414 * 0.307 - 10d.ToRad()
    , "YZX");

    return new Dictionary<string, EulerAngles>()
    {
        { "SternoClavicularLeft", sternoClavicularLeftDOFs },
        { "SternoClavicularRight", sternoClavicularRightDOFs },
        { "ScapulaThoraxLeft", scapulaThoraxLeftDOFs },
        { "ScapulaThoraxRight", scapulaThoraxRightDOFs }
    };
}
```

Spine Rhythm Implementation

```
public static Dictionary<string, EulerAngles> SpineRhythmDOFs(EulerAngles pelvisThoraxDOFs)
{
    pelvisThoraxDOFs = pelvisThoraxDOFs.ToEulerAngles("ZYX");

    var t12L1WeightMatrix = Matrix<double>.Build.DenseOfArray(new[,]
    {
        // Matrix from spine rhythm definition in file /Body/AAUHuman/Trunk/SRMatrixes.any
        { 7.105616e-002, 2.276759e-001, 4.020500e-001, 5.784718e-001, 7.462112e-001, 9.131695e-001, 1.00000000000 }, // X factors
        { 0.00000000000, 1.421123e-001, 3.132395e-001, 4.908604e-001, 6.660833e-001, 8.263391e-001, 1.00000000000 }, // Y factors
        { 7.105616e-002, 2.276759e-001, 4.020500e-001, 5.784718e-001, 7.462112e-001, 9.131695e-001, 1.00000000000 }  // Z factors
    });
    var t12L1DOFs = new EulerAngles(pelvisThoraxDOFs.X / t12L1WeightMatrix.Row(0).Sum(), pelvisThoraxDOFs.Y / t12L1WeightMatrix.Row(1).Sum(),
pelvisThoraxDOFs.Z / t12L1WeightMatrix.Row(2).Sum());

    return new Dictionary<string, EulerAngles>()
    {
        { "SacrumPelvis" , new EulerAngles(t12L1DOFs.X * t12L1WeightMatrix[0, 0], t12L1DOFs.Y * t12L1WeightMatrix[1, 0], t12L1DOFs.Z *
        t12L1WeightMatrix[2, 0]) },
        { "L5Sacrum"     , new EulerAngles(t12L1DOFs.X * t12L1WeightMatrix[0, 1], t12L1DOFs.Y * t12L1WeightMatrix[1, 1], t12L1DOFs.Z *
        t12L1WeightMatrix[2, 1]) },
        { "L4L5"         , new EulerAngles(t12L1DOFs.X * t12L1WeightMatrix[0, 2], t12L1DOFs.Y * t12L1WeightMatrix[1, 2], t12L1DOFs.Z *
        t12L1WeightMatrix[2, 2]) },
        { "L3L4"         , new EulerAngles(t12L1DOFs.X * t12L1WeightMatrix[0, 3], t12L1DOFs.Y * t12L1WeightMatrix[1, 3], t12L1DOFs.Z *
        t12L1WeightMatrix[2, 3]) },
        { "L2L3"         , new EulerAngles(t12L1DOFs.X * t12L1WeightMatrix[0, 4], t12L1DOFs.Y * t12L1WeightMatrix[1, 4], t12L1DOFs.Z *
        t12L1WeightMatrix[2, 4]) },
        { "L1L2"         , new EulerAngles(t12L1DOFs.X * t12L1WeightMatrix[0, 5], t12L1DOFs.Y * t12L1WeightMatrix[1, 5], t12L1DOFs.Z *
        t12L1WeightMatrix[2, 5]) },
        { "T12L1"        , t12L1DOFs }
    };
}
```