2019 IEEE International Conference on Advanced Robotics and its Social Impacts (ARSO)
BIT, Beijing, China. October 31 - November 2, 2019

Musculoskeletal Estimation Using Inertial Measurement Units and Single Video Image

Vincent Samy, Ko Ayusawa, Yusuke Yoshiyasu, Ryusuke Sagawa and Eiichi Yoshida[1]

Abstract— **We address the problem of estimating the physical burden of a human body. This translates to monitor and estimate muscle tension and joint reaction forces of a musculoskeletal model in real-time. The system should minimize the discomfort generating by any sensors that needs to be fixed on the user. Our system combines a 3D pose estimation from vision and IMU sensors. We aim to minimize the number of IMU fixed to the subject while compensating the remaining lack of information with vision.**

I. Introduction

Not all factories are automatized and humans will keep working in this environment. One of the main reason is that robots have still problems to adapt to the human environment. In opposite, humans navigate easily in any environment, however, several studies [1][2] have shown that factory workers face muscular fatigue and physical health difficulties on the long term. One of the focus of this paper is low back (lumbar) pain which is one of the main problem encountered by workers. The National Institute of Occupational Safety and Health (NIOSH) has set a maximum limit load on the inter-vertebral discs [3] which is often used to qualify a back pain risk. Monitoring in real-time this load is a complex operation, first because its computation is not trivial and second because the used system must not be cumbersome for the worker. The latter excludes intrusive methods like [4] which provide a better precision results but are impracticable. The former goes with estimations (and thus imprecision), and with some time constraints since all the computations need to be done in a short amount of time. Although several off-line estimations exist [5][6], the focus will be made on on-line computation while reducing cumbersomeness.

In our previous work [6], we developed a framework of visualizing the physical burden of human body during movements by using the human musculoskeletal model [7]. The system can realize the real-time estimation of the several information like joint angles, joint torques, muscle tensions, and joint reaction forces [5]. The main objective is to support factory workers by monitoring the risk of physical health problems like low back pains. However, the system requires a large amount of Inertial Measurement Unit (IMU) sensors in order to obtain the accurate results, which reduces the workers' comfort during their tasks.

*This work was partly supported by JSPS KAKENHI Grant No. 17H00768, No. 18H03315, and No.17K18420.

[1]V. Samy, K. Ayusawa, Y. Yoshiyasu, R. Sagawa and R. Sagawa are with CNRS-AIST JRL (Joint Robotics Laboratory), UMI3218/RL, Tsukuba, Ibaraki, Japan. Corresponding Author: V. Samy vincent.samy@aist.go.jp

Recently, Ohashi *et al.* [8] presented an algorithm based on 4 camera that computes Human pose by using a spatiotemporal filter. They could reach a precision of less than 3cm in average. While the precision is remarkable, the system confines the subject in a space where all cameras need to see him. IMU-based systems also exist, Marcard *et al.* [9] suggest a method that merges a statistical body model that includes anthropometric constraints with 6 IMU sensors. Malleson *et al.* [10] proposed a method to get the 3D human pose estimation from multiple cameras and from 6 to 13 IMUs in real-time. Methods as in [11] uses the deep neural network to train both video images and the data of IMU sensors. This approaches have a benefit of integrating several inputs of data resources easily. The video cameras can be introduced in a factory and will provide supplementary information about human movement. On the other hand, they are not always available during whole period of working. Therefore, the pose estimation requires the flexibility of changing the inputs of data resources according to situations. In addition, the straightforward pose estimation often lacks the human musculoskeletal feasibility like joint or muscle-tendon constraints.

In this paper, we propose a new framework of musculoskeletal estimation by utilizing the vision-based pose estimation technique with deep learning [12]. Therefore, our method is based on the motion optimization with the musculoskeletal model to integrate the pose information obtained from IMU sensors and vision images. The musculoskeletal optimization can provide both the human feasibility and the flexibility about data integration. Based on the fast robotics kinematics computation [13], [14], [15], our method also realize the real-time computation of the detailed human musculoskeletal model [7], By introducing the vision-based pose estimation, the number of IMU sensors can be reduced while keeping the accuracy of musculoskeletal estimation. Our method does not use the data of IMU sensors in the training of the deep neural network, which also enables the flexible change of the placement of IMU sensors according to applications. This paper also show several experimental validations about the accuracy by testing the several configurations of sensor placements.

II. Method

We propose a system that estimates musculoskeletal activities from different set of sensors.

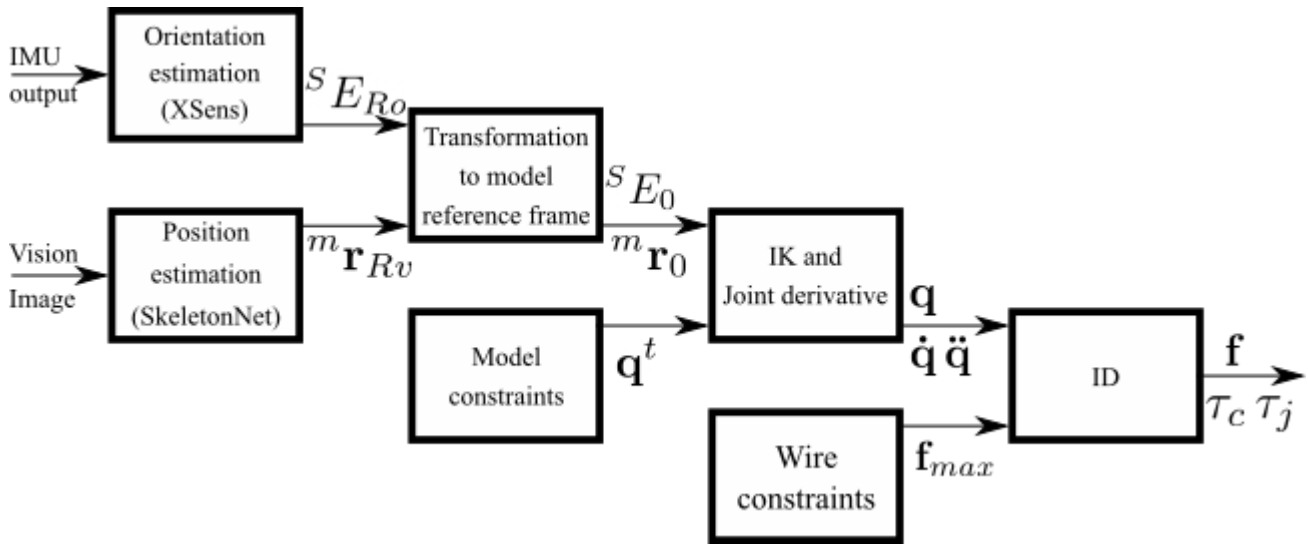


Fig. 1: Process from sensors outputs to wire tension and joint reaction forces computation. It takes as input IMUs and an image. XSens SDK provide IMUs' orientation ${}^{S}E_{Ro}$ and SkeletonNet provides joint positions ${}^{m}\mathbf{r}_{Rv}$, both are respectively transform in world coordinates to get ${}^{S}E_{0}$ and ${}^{m}\mathbf{r}_{0}$. Then, they are sent to the IK along model constraints $\mathbf{q}^{t}$. Output of the IK $\mathbf{q}$ is derived and filtered to get $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$. Finally, data are sent to the ID along wire constraints $\mathbf{f}_{\text{max}}$ to output the wire tension $\mathbf{f}$, the joint torque $\boldsymbol{\tau}_j$ and the joint reaction force $\boldsymbol{\tau}_c$.

A. Overview

The flow of the system (see Fig. 1) works like this: i) get sensors' data, ii) Transform data into the musculoskeletal model reference frame, iii) compute the Inverse Kinematics (IK), and iv) feed the Inverse Dynamics (ID) with IK results.

We used two sets of sensors. For body orientation, 12 IMU sensors[1] are attached to wrists, shoulders, ankles, thighs, head, torso and back, and for body position, a webcam[2] visualizes the scene. We use the XSens provided SDK to collect IMU outputs and SkeletonNet [12] system to get the 3D pose estimation.

SkeletonNet uses a single camera with depth and deep learning methods to return joint position in the world frame. The system uses a two-step regressor (see Fig. 2) that first performs a bone rotation regression then a cross heatmap regression. Note that the regressor used here is less complex than the one presented in [12].

Although IMU sensors don't need any model to work, the 3D pose estimator is based on a simplified human model composed of 16 points that correspond to the ankles, knees, hips edges, hip center, neck, shoulders, elbows, hands, lower and upper head positions. This model is rather simple and thus, is different from the one in the IK and ID. The IK/ID model is composed of 14 joints, which correspond to 47 degrees of freedom, 314 muscles, 6 tendons and 34 cartilages.

[1]XSens awinda series: `https://www.xsens.com/products/mtw-awinda/`

[2]Logitech HD C615: `https://www.logitech.com/en-us/product/hd-webcam-c615`

B. Musculoskeletal variables estimation

To synchronize XSens with the IK model, and SkeletonNet with the IK model, we added an initialization step which allow us to get correct transformations from measurement to model. At the initialization step, the subject is asked to match an upright configuration with arms along the body Fig. 4. From this configuration we get both ${}^{Ro}E_{S}^{i}$ the rotation of the i-th sensor from its current frame $\mathcal{F}_S$ to its reference frame $\mathcal{F}_{Ro}$ and ${}^{m}\mathbf{r}_{Rv}^{k}$ the k-th marker position from its reference frame $\mathcal{F}_{Rv}$ to the marker position m. For XSens, we need to know the rotation from the IMU reference frame and the orientation of the arm it is attached to. Let's have ${}^{G}E_{b}$ the Given rotation from the body orientation to the attached sensor. This value is decided beforehand and is chosen so that the IMU is easy to place on the body (like a flat surface). Provided that the frame $\mathcal{F}_G$ and the frame $\mathcal{F}_S$ should match, we compute the transformation

$$ {}^{Ro}E_{0}^{i} = {}^{Ro}E_{S}^{i}\,{}^{G}E_{b}^{i}\,{}^{b}E_{0}^{i} \tag{1} $$

with ${}^{b}E_{0}^{i}$ the orientation of body i in the world frame. We also compute the offset from $\mathcal{F}_G$ to $\mathcal{F}_S$ with

$$ {}^{S}E_{G}^{i} = {}^{S}E_{Ro}^{i}\,{}^{Ro}E_{0}^{i}\,{}^{0}E_{G}^{i}. \tag{2} $$

In the same manner, we define

$$ {}^{b}\mathbf{r}_{m}^{i} = {}^{b}\mathbf{r}_{0}^{i} - {}^{0}E_{Rv}\,{}^{m}\mathbf{r}_{Rv}^{i} \tag{3} $$

where ${}^{0}E_{Rv}$ is the rotation form SkeletonNet reference frame to the world frame and ${}^{b}\mathbf{r}_{0}^{i}$ the position of body i in the world frame.

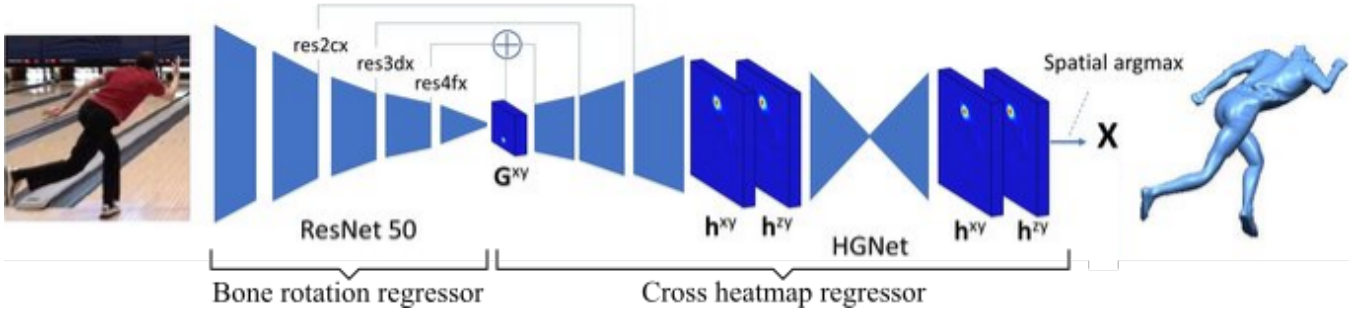


Fig. 2: SkeletonNet system that outputs joint positions using a two-step regressor.

Now that data are transformed into the same frame, we can compute the IK:

$$\min_{q} \ \omega_q \|\mathbf{q}_t^i - \mathbf{q}^i\|^2 + \omega_r \|{}^t\mathbf{r}_0^i - {}^b\mathbf{r}_0^i\|^2 + \omega_E \|{}^tE_0^i - {}^bE_0^i\|^2 \quad (4)$$

with $\mathbf{q}^i$ the generalized coordinates of angle i and $\mathbf{q}_t^i$ its target, ${}^b\mathbf{r}_0^i$ the position of body b in the world frame and ${}^t\mathbf{r}_0^i$ its target, ${}^bE_0^i$ the orientation of body b in the world frame and ${}^tE_0^i$ its target, $\omega_k, k = \{q, r, E\}$ the weight. The vision system provides markers as position markers relative to the camera and IMU sensors provide body orientation. When the subject performs a whole body rotation, the IK has difficulties to rotate the root body at the same amount. This is because the pelvis (*e.g.* the root body), has no parent joint (it is freely attached to the world). To prevent this, we add a special task which target a specific joint angle value for all joints that link a body to the pelvis. We also add tasks that acts as joint limit constraints. These values are set at the initialization step and are not changed afterwards.

Last point is the synchronization of data. SkeletonNet returns data every 16ms in average but varies from 14ms up to 22ms for worst cases. XSens returns values every 16ms. We decided to have a loop that runs every 30ms. The loop first get data from both SkeletonNet and XSens (using protective methods like mutex to ensure data validity). Then it computes the IK. The IK can either end because a limit of time is reached or if converged. The derivative of IK's result are computed and filtered and then sent to the ID.

The optimization system to compute the inverse dynamics is [5]:

$$\begin{cases} \min\limits_{\mathbf{f},\boldsymbol{\tau}_j,\boldsymbol{\tau}_c} & Z_f(\mathbf{f}) + Z_j(\mathbf{f},\boldsymbol{\tau}_j) + Z_c(\mathbf{f},\boldsymbol{\tau}_c) \\ \text{s.t.} & \text{-}\mathbf{f}_{\max} \leq \mathbf{f} \leq \mathbf{0} \end{cases} \quad (5)$$

with

$$\begin{aligned} Z_f &= \mathbf{f}^{\mathsf{T}} W_f \mathbf{f} \\ Z_k &= (\boldsymbol{\tau}_k - J_k^{\mathsf{T}}\mathbf{f})^{\mathsf{T}} W_k (\boldsymbol{\tau}_k - J_k^{\mathsf{T}}\mathbf{f}), k = j, c, \end{aligned} \quad (6)$$

where $\mathbf{f} \in \mathbb{R}^{\mathrm{N_w}}$ is the wire tension, $\boldsymbol{\tau}_j \in \mathbb{R}^{\mathrm{N_{dof}}}$ is the joint torques, $\boldsymbol{\tau}_c \in \mathbb{R}^{\mathrm{N_c}}$ is the joint reaction forces, $J_j \in \mathbb{R}^{\mathrm{N_w} \times \mathrm{N_{dof}}}$ is the Jacobian matrix that maps the joint torques to the wire tension, $J_c \in \mathbb{R}^{\mathrm{N_w} \times \mathrm{N_c}}$ is the Jacobian matrix that maps the joint reaction forces to the wire tension, and W_f, W_j and W_c are weighting matrix.

III. EXPERIMENT

We conducted experiments in order to compare the results of estimated physical quantities among several configurations of input data resources. Though our framework utilizes IMUs and a single video camera, the optical motion capture system (Motion Analysis Corp.) was also used in order to be compared.

Our aim is not only to reduce the number of IMUs but also to know which IMUs should be used. To compare different IMU configuration, we decided to record a complete data-set with all systems. As in Fig. 3, the subject wears a suit with 34 motion capture markers and 12 IMUs. The vision system is composed of one camera (see Fig. 4) set in front of the subject. All data are streamed to the main computer that record data altogether so that all are synchronized. Although it is possible to compute wire tension and joint reaction forces in a dynamic way, to simplify the comparison, we decided to limit the computation of the ID at the static level, leaving $\dot{\mathbf{q}} = \mathbf{0}$ and $\ddot{\mathbf{q}} = \mathbf{0}$.

Fig. 3: Sensor positioning. The suit is composed of 34 motion capture markers and 12 IMUs.

During the recording, the subject is asked to perform several movement with different facings (facing the camera or not). Some of the movement includes step, throw, lift, crouch, sit, radio gymnastic, etc.

IV. RESULTS

Among all different motion we will show some results on 4 different level. Fig. 5a and 5b is about the joint angle (x-axis) of the left shoulder during lifts. Fig. 6a and 6b compares joint angle (y-axis) of the lumbar vertebra 5 during steps

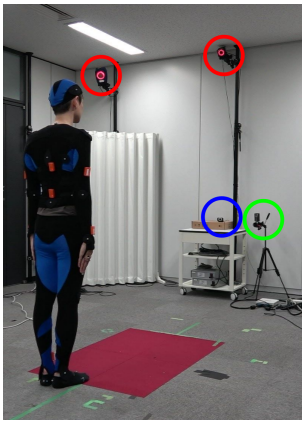


Fig. 4: Experiment initialization. Red circle shows motion capture cameras, blue circle shows the camera used for the vision, green for the recording.

and throws. Fig. 7a and 7b shows joint reaction force (y-axis) of the lumbar vertebra 5 during crouches. And finally Fig. 8a and 8b represents data of the rectus femoris muscle during crouches and sit. For each motion (step, crouch, etc), the subject performs it facing, siding and backing on to the camera. All the computation process has been made in real-time at a rate of 33Hz on a single computer (Intel Xeon Gold 6134 CPU @3.20 GHz and NVIDIA Quadro P200).

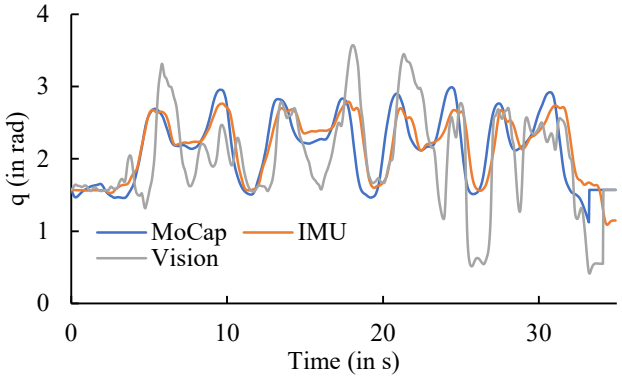
Here, we compare different set of IMU disposition on the body along the vision system. In the following graph we used *Ext* for 5 IMUs on chest, wrists and ankles, *Int* for 5 IMUs on chest, shoulders and thighs, *Mix5* for 5 IMUs on chest, wrists and thighs, and *Mix6* the same as *Mix5* with one more IMU on the lumbar vertebra 1.

Fig. 5a shows that the vision system does not yet have a precision good enough to run on its own, although it sometimes can perform as good as a full IMU set, it can not maintain the precision. This is due to the occlusion that might happen and the difficulties for the estimator to get the 3D pose from dynamic motion that generates blurry images. Fig. 5b compares different IMUs disposition. It turns out that the best disposition in this case is the *Int*. It almost always follow the MoCap value which is consider of the closest to reality.

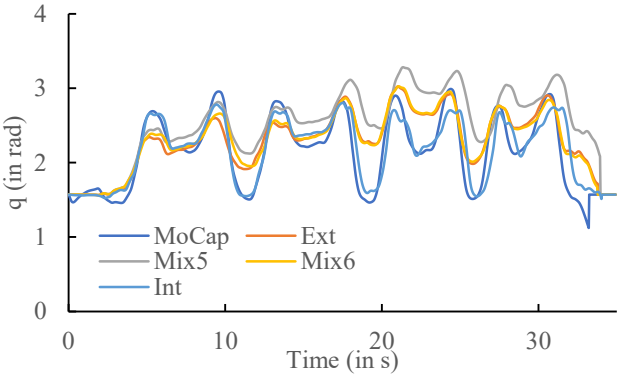
To prevent and alert potential back pain we need to focus on lumbar forces. We now consider a joint belonging to the trunk and that is used to compute these forces. In that case, Fig. 6a, the vision system can not see how the person is bending since it only return the hip and chest position. Of course, having a sensor close to the lumbar vertebra (line *Mix6* in Fig. 6b[1]) provides a better solution than others.

We can see in Fig. 8a and 8b the impacts of these errors on the lumbar vertebra joint reaction forces. Vision system can not output any correct data while the XSens system can at least provide intensity spikes. Mixing these data with IMUs

[1]The results are subject to small time delay that increase slightly through time.



(a) X-axis of spherical joint of the left shoulder during lifts motion for MoCap, IMU and vision methods alone.



(b) X-axis of spherical joint of the left shoulder during lifts motion for MoCap and several mix methods combining vision and IMUs.

Fig. 5: Joint angle visualization of the left shoulder during lifts motion.

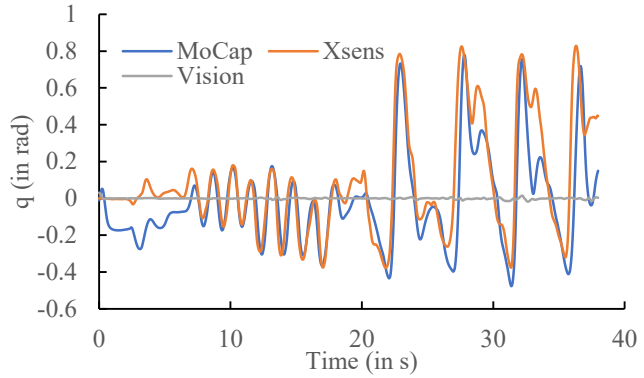
return Fig. 7b. It shows that adding a sixth IMU on the lumbar vertebra also greatly increase the system accuracy in term of joint reaction force. Note that the *Ext* provides better results than other methods but it can not catch sudden spike intensities.

Finally, we also looked at a muscle tension, the rectus femoris, that links the pelvis to the femur Fig. 8a and 8b. Again, regarding the subject posture, the vision system sometimes fails matching the 3D pose. Here, the *Ext* disposition gives the worst results, it is also the only one that does not have IMU on its thighs.

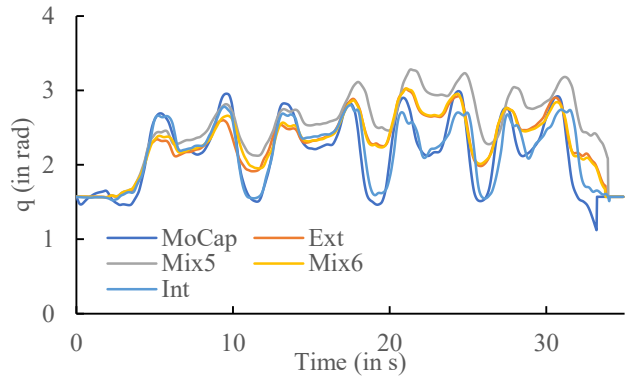
Globally using vision allows to reduce the number of IMUs while keeping accuracy. For case of the spine, it is necessary to attach several IMUs on it. Another solution would be to have a better model for the vision. For dynamic computation, it is better to have IMUs close to muscle tension/joint reaction force you want to have results on.

V. Conclusion

We proposed a new framework of musculoskeletal estimation by using IMUs and a single video camera. The motion optimization integrates the input data resources with keeping the accuracy of musculoskeletal estimation. By integrating the vision-based pose estimation, we showed that the number

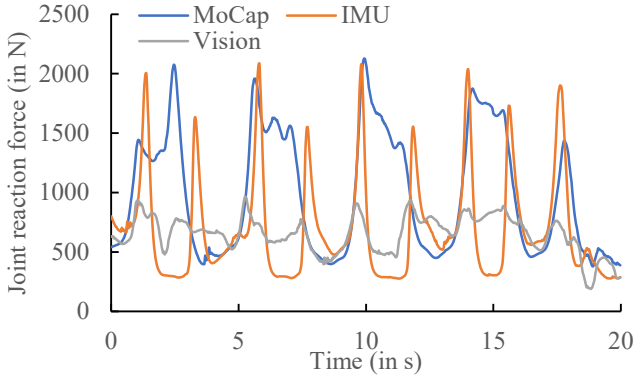


(a) Y-axis of spherical joint of the lumbar vertebra 5 during steps and throws motion for MoCap, IMU and vision methods alone.

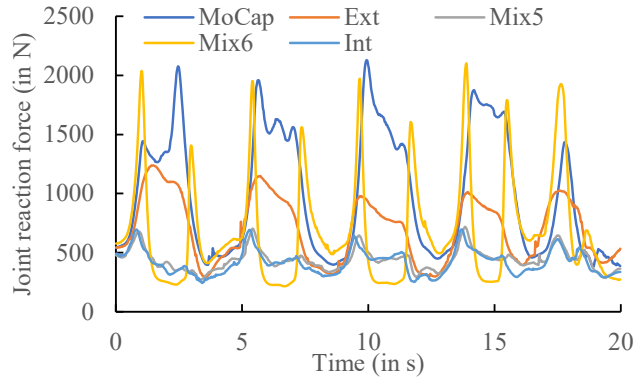


(b) Y-axis of spherical joint of the lumbar vertebra 5 during steps and throws motion for MoCap and several mix methods combining vision and IMUs.

Fig. 6: Joint angle visualization of the lumbar vertebra 5 during steps and throws motion.



(a) Y-axis of joint reaction force of the lumbar vertebra 5 during throws motion for MoCap, IMUs and vision methods alone.



(b) Y-axis of joint reaction force of the lumbar vertebra 5 during throws motion for MoCap and several mix methods combining vision and IMUs.

Fig. 7: Joint reaction force of the lumbar vertebra 5 during steps and throws motion.

of IMU sensors attached to a subject can be reduced. The consistency between two time instances is also preserved thanks to the IMUs sensors, while the only vision-based pose estimation often loses the consistency. The method can provide a flexible IMU disposition so that it can be adjusted depending on which body segment we want to analyze accurately; for example, IMUs on the trunk would allow the accurate estimation of the joint reaction forces of the lumber-vertebra. Therefore, our method is expected to be used to monitor the back pain risk of a individual subject in an actual working environment.

VI. Future work

Here, we discuss a way to reduce the number of IMU without loosing too much precision. A next job would be to use multiple camera and to be robust with lost signals (out-of-range IMU or occlusion).

The benefits of using vision is twofold, it is less cumbersome for the subject and, in the future, we can exploit it to get the global pose and even detect contacts with the environment. And to better exploit vision capabilities we do need to update our vision system to have a more robust system. For the sake of accuracy, we also want to include dynamic consistency inside the optimization process so that wire tension and joint reaction forces would return better results.

In a further future, we would like to use the software on site, to monitor workers. Doing so, we hope that we will be able to reduced the amount of physical pains and improve the workers' work life.

References

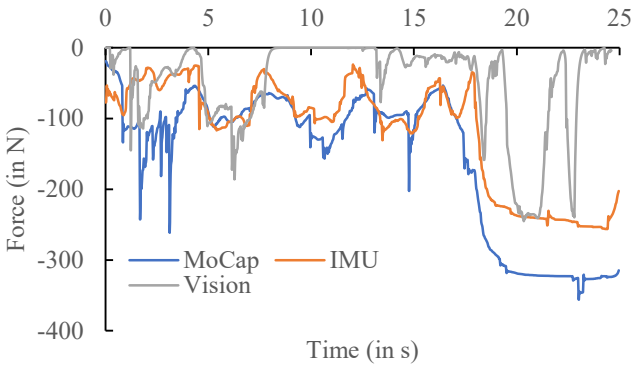
[1] B. Duthey, "Background paper 6.24 low back pain," *Priority medicines for Europe and the world. Global Burden of Disease (2010),(March)*, pp. 1–29, 2013.

[2] L. Punnett, A. Prüss-Ütün, D. I. Nelson, M. A. Fingerhut, J. Leigh, S. Tak, and S. Phillips, "Estimating the global burden of low back pain attributable to combined occupational exposures," *American journal of industrial medicine*, vol. 48, no. 6, pp. 459–469, 2005.

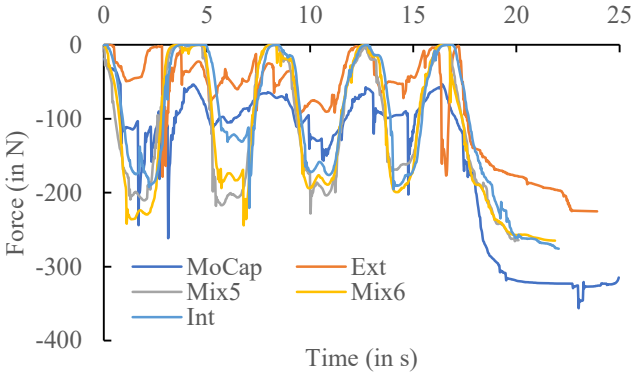
[3] T. R. Waters, V. Putz-Anderson, A. Garg, and L. J. Fine, "Revised niosh equation for the design and evaluation of manual lifting tasks," *Ergonomics*, vol. 36, no. 7, pp. 749–776, 1993.

[4] H.-J. Wilke, P. Neef, B. Hinz, H. Seidel, and L. Claes, "Intradiscal pressure together with anthropometric data–a data set for the validation of models," *Clinical Biomechanics*, vol. 16, pp. S111–S126, 2001.

[5] Y. Imamura, K. Ayusawa, and E. Yoshida, "Risk estimation for intervertebral disc pressure through musculoskeletal joint reaction force simulation," in *Engineering in Medicine and Biology Society (EMBC), 2017 39th Annual International Conference of the IEEE*. IEEE, 2017, pp. 1636–1639.



(a) Rectus femoris muscle force during crouches and sit motion for MoCap, IMUs and vision methods alone.



(b) Rectus femoris muscle force during crouches and sit motion for MoCap and several mix methods combining visions and IMUs.

Fig. 8: Rectus femoris muscle force during crouches and sit motion.

[6] V. Samy, K. Ayusawa, and E. Yoshida, "Real-time musculoskeletal visualization of muscle tension and joint reaction forces," in *2019 IEEE/SICE International Symposium on System Integration (SII)*, Jan 2019, pp. 396–400.

[7] Y. Nakamura, K. Yamane, Y. Fujita, and I. Suzuki, "Somatosensory computation for man-machine interface from motion-capture data and musculoskeletal human model," *IEEE Transactions on Robotics*, vol. 21, no. 1, pp. 58–66, 2005.

[8] T. Ohashi, Y. Ikegami, K. Yamamoto, W. Takano, and Y. Nakamura, "Video motion capture from the part confidence maps of multi-camera images by spatiotemporal filtering using the human skeletal model," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2018, pp. 4226–4231.

[9] T. Marcard, B. Rosenhahn, M. Black, and G. Pons-Moll, "Sparse inertial poser: Automatic 3d human pose estimation from sparse imus," *Computer Graphics Forum 36(2), Proceedings of the 38th Annual Conference of the European Association for Computer Graphics (Eurographics), 2017*, vol. 36, 02 2017.

[10] C. Malleson, A. Gilbert, M. Trumble, J. Collomosse, and A. Hilton Cvssp, "Real-time full-body motion capture from video and imus," 10 2017.

[11] M. Trumble, A. Gilbert, C. Malleson, A. Hilton, and J. Collomosse, "Total capture: 3d human pose estimation fusing video and inertial sensors," 09 2017.

[12] Y. Yoshiyasu, R. Sagawa, K. Ayusawa, and A. Murai, "Skeleton transformer networks: 3d human pose and skinned mesh from single rgb image," *Lecture Notes in Computer Science*, p. 485–500, 2019. [Online]. Available: http://dx.doi.org/10.1007/978-3-030-20870-7_30

[13] A. Murai, K. Kurosaki, K. Yamane, and Y. Nakamura, "Musculoskeletal-see-through mirror: Computational modeling and algorithm for whole-body muscle activity visualization in real time." *Progress in Biophysics and Molecular Biology*, vol. 103, no. 2-3, pp. 310–317, 2010.

[14] K. Ayusawa and Y. Nakamura, "Fast inverse dynamics algorithm with decomposed computation of gradient for wire-driven multi-body systems and its application to estimation of human muscle tensions," in *2nd IFToMM International Symposium on Robotics and Mechatronics (11)*, 2011.

[15] ——, "Fast inverse kinematics algorithm for large dof system with decomposed computation of gradient and its application to musculoskeletal model," in *Proc. of the 17th Robotics Symposia*, 2012, pp. 148–155.