Whole-Body Sensory Concept for Compliant Mobile Robots

Marina Kollmitz Daniel Büscher Tobias Schubert Wolfram Burgard

***Abstract*— Most of the conventional approaches to mobile robot navigation avoid any kind of contact with the environment or with humans. As nowadays distance sensors typically have a limited – and often only two-dimensional – field of view, collisions with the environment or contacts with humans cannot be fully avoided in practical mobile robot applications. On the other hand, direct physical contact can be used for intuitive communication between a robot and humans. In this paper, we present a whole-body sensory concept based on a 6-DoF force-torque sensor to perceive physical interaction between the robot and humans. To distinguish between external contact and disturbance forces that result from the motion of the mobile platform or oscillations, we present a novel model-free filtering approach based on a neural network. In extensive experiments carried out with our robot Canny we demonstrate the effectiveness and advantages of the neural network approach, which clearly outperforms a classical model-based one.**

I. Introduction

Robots are envisioned to increasingly share their workspaces with humans, which requires new concepts for human-robot interaction. The close proximity allows for intuitive and efficient physical interaction. One possible interaction scenario is shown on the left in Fig. 1, where a person pushes the robot out of her desired path. The approach presented in this paper tackles this interaction scenario by enabling the robot to feel interaction forces during autonomous motion.

In this paper we present a novel whole-body sensory concept based on a six degrees of freedom (DoF) force-torque sensor mounted stiffly on an omnidirectional mobile base, as shown in the right image of Fig. 1. A solid outer shell attached to the other side of the sensor absorbs and directs external forces to the sensor. Our setup enables the robot to feel the extend, direction and location of interaction forces on the robot shell. Even though the motion capabilities of wheeled mobile robots are limited to the 2D plane, knowledge of the 3D direction and impact of the force are valuable to distinguish different types of interactions. An intended contact is most likely occurring in reach of a person's hands, while a collision can occur anywhere on the robot. A person leaning against the robot likely causes a force with a sideways and downwards component. We think the robot should adapt its behavior to different kinds of contact, and the force characteristics can help distinguish them. For multiple contacts, our setup can only process the resulting force but cannot discriminate between them. We accept this limitation, since the resulting force already provides valuable information for the envisioned contact scenarios amongst a crowd of people.

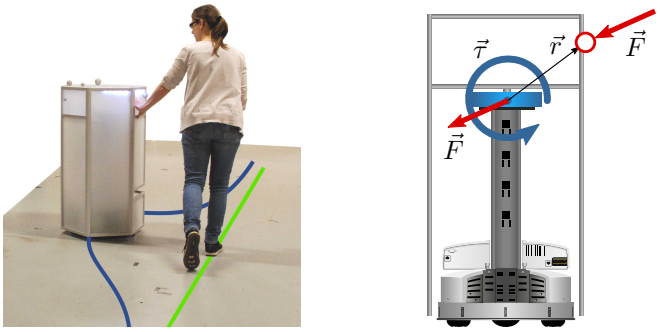

Fig. 1. The presented sensory concept is based on a 6-DoF force-torque sensor and enables the robot to perceive interaction forces.

A more severe downside of the sensory concept is the sensitivity to force stimuli caused by the robot motion. Furthermore, since the connections between robot base, force-torque sensor and shell are not perfectly stiff, the system is oscillatory. The induced internal forces superimpose the true external interaction forces. We present two filtering techniques to distinguish the resulting disturbance forces from external forces. The first, probabilistic approach uses a physical model of the base response to external and internal force stimuli. It serves as reference for our main approach, which employs end-to-end learning using a neural network with a multi-task output. The sensory concept combined with the filtering technique enables us to precisely estimate the magnitude, direction and location of an external force, even when the robot is in motion. In extensive experiments we evaluate the performance of our system. To this end, we present a novel experimental setup where varying external forces of known magnitudes, directions and impact points can be applied to any point of the robot shell.

We believe that our design is an important contribution along the way to safe, interactive mobile robots that operate in close proximity to humans. In addition, force-torque sensors like the one we use for our robot Canny are standard components that are widely applied in robotics. Thus, our design is easily applicable to other robot platforms.

II. Related Work

Autonomous robot navigation amongst humans is a challenging problem, since the shared environments are typically cluttered, highly dynamic and hardly predictable. Especially in crowded environments, it is necessary to assume that all agents cooperate in order to make progress towards the

*This work has been partially supported by the German Federal Ministry of Education and Research (BMBF), contract number 01IS15044B-NaRKo.

The authors are with the Department of Computer Science, University of Freiburg, Freiburg, Germany. {kollmitz, buescher, tobschub, burgard}@informatik.uni-freiburg.de

goal [1, 2]. Prassler et al. presented a robot wheelchair that used the Velocity Obstacle approach [4] for navigation in crowded environments. The Minerva museum tour guide robot [5] reacted with angry voices and expressions on its actuated face to eke out a path. Our approach seeks to extend such existing navigation approaches by enabling the robot to perceive interaction forces and thus to include and react to intentional physical contact and accidental collisions.

Work on force-controlled robots dates back to the 1990s when Khatib [6] introduced the robotic assistant for cooperative object manipulation between humans and mobile manipulators. Later, Haddadin et al. [7] presented a concept for human safety in shared workspaces for robot manipulators. Hirata et al. [8] proposed an approach to cooperative object transportation where a human pushed the robot along a preplanned path, indicating the desired robot velocity. Similar to us, they used a 6-DoF force-torque sensor between the robot base and the shell to measure interaction forces. However, they did not evaluate the impact location of the force.

Walking helper systems for the elderly have been presented that react to a force exerted by the user, measured e.g., by strain gauges [9] or 6-DoF force-torque sensors [10, 11]. The design of the devices is similar to ours in that a force sensor adds sensitivity to a mounted stiff structure. However, unlike our robot shell, not all exposed parts of the walking devices were force sensitive because the mounted structure did not cover the entire devices. Furthermore, the motion of the walking helpers was always a reaction to intended force. Instead, we strive to combine autonomous motion with a compliant reaction to interaction forces.

Manuelli and Tedrake [12] used a particle filter to estimate the contact points of external forces on rigid body humanoid robots from the joint torques. Their approach was able to accurately estimate the locations of multiple impact points for a simulated humanoid robot. Kim et al. [13] also estimated the external forces based on the joint torques of their mobile base with torque sensors in the drive trains of the three omnidirectional wheels. Their robot could react to interaction forces and detect collisions with the entire robot body, in rest and in motion. However, they only estimated the position and direction of the external force in the horizontal plane due to the limited degrees of freedom of the robot. Furthermore, they did not evaluate the location, direction and magnitude of the external force when the robot was in motion. Frémy et al. [14] also used the torques on the caster wheel joints of their mobile robot to estimate external forces. However, their platform was not fully omnidirectional, which limited the possible reactions to interaction forces.

Our work uses the approach of Bicchi et al. [15] to determine the impact point of an applied force. We extend their work by filtering out disturbance forces caused by the motion of the robot and oscillations of its shell. Thus, we render their approach to real mobile robot applicability.

III. Whole-Body Sensory Concept

Our robot Canny is based on the omnidirectional research platform Robotino. The Robotino is shipped with a mounting tower that attaches tightly to the base. We mount a high-precision 6-DoF force-torque sensor to the tower as depicted in Fig. 1. The other side of the force-torque sensor is attached to a solid robot shell, made of aluminum profiles and semi-transparent acrylic glass. An inertial measurement unit (IMU) at the robot base measures the base accelerations in the six DoFs. Furthermore, colored LEDs at the rim of the robot shell display user feedback, such as the direction and magnitude of the perceived interaction force.

We calculate the impact point of an external force from the measured forces and torques according to Bicchi et al. [15]. We restrict ourselves to the model of a *point contact with friction*, which means that the force is applied on a single point instead of an area. Accordingly, we neglect local torques during the impact. For convex shell geometries, there is a unique point of impact for a force, assuming that it results from pushing on the robot shell instead of pulling.

For an impact force $\vec{F}$ on the robot shell at a lever arm $\vec{r}$ (Fig. 1), the force-torque sensor measures the force $\vec{F}$ and a corresponding torque $\vec{\tau}$. We approximate Canny's hexagonal shell by a cylinder with radius R and axis $\vec{z} \in S^2$ and obtain the impact point according to Bicchi et al. [15], as

$$\vec{r} = \vec{r}_0 + \lambda \vec{F} \text{ where } \vec{r}_0 = \frac{\vec{F} \times \vec{\tau}}{\|\vec{F}\|} \tag{1}$$

and

$$\lambda = \frac{-\vec{F}^{\perp} \cdot \vec{r}_0^{\perp} - \sqrt{(\vec{F}^{\perp} \cdot \vec{r}_0^{\perp})^2 - \|\vec{F}^{\perp}\|^2 (\|\vec{r}_0^{\perp}\|^2 - R^2)}}{\|\vec{F}^{\perp}\|^2}. \tag{2}$$

Thereby, $\vec{v}^{\perp} = \vec{v} - (\vec{v} \cdot \vec{z})\,\vec{z}$ denotes the part of a vector $\vec{v} \in \mathbb{R}^3$ which is orthogonal to $\vec{z}$.

IV. External Force Filtering

We present two approaches to separate the external impact forces from disturbance forces caused by robot motion and oscillations. First, we introduce a probabilistic, model-based approach in Sec. IV-A. Then, in Sec. IV-B, we present a model-free method based on a neural network.

A. Model-Based External Force Estimation

We model our system as a stiff body suspended on a system of springs and dampers. As a simplification, we assume that the robot structure, the shell and the mounting tower are perfectly stiff, while the force-torque sensor is the sole source of elasticity. We use the coordinate system of the sensor, where the z-axis points vertically up, and we model the kinematics of the system in six degrees of freedom.

To derive the equations of motion, we first consider the 1D case of an oscillatory system as depicted in Fig. 2 (left). The mass m is connected to a spring k and a damper c. The force F acting on the mass comprises of an external force F_{ext} and a force $F_a = -ma$ caused by the acceleration a of the inertial frame, i.e., the acceleration of the robot. The deflection of the spring is denoted by q. Thus, for the equation of motion we have

$$m\ddot{q} + c\dot{q} + kq = F_{\text{ext}} - ma. \tag{3}$$

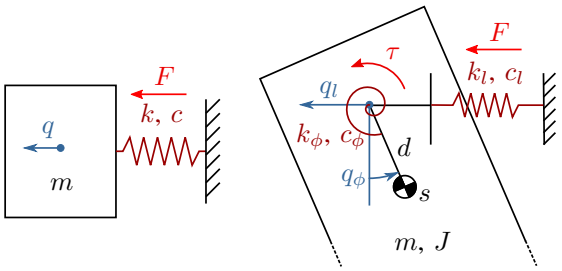

Fig. 2. One-dimensional linear spring-mass system (left) with spring constant k and damping constant c. On the right, our extended 2D case with displaced center of mass, resulting in coupled linear and angular oscillations. The walls in the drawings correspond to the robot base, which itself is movable, resulting in additional accelerations of the reference frame.

If the robot shell is mounted at its center of mass, the degrees of freedom are decoupled and can be described by Eq. (3). However, this does not hold in our case, as the mount of the shell is displaced vertically along the z-axis. Hence, some degrees of freedom are coupled, as depicted in Fig. 2 (right), where we consider the 2D case of coupled linear oscillation in q_l and angular oscillation in q_ϕ. The stiff robot shell with mass m and moment of inertia J is mounted at distance d from its center of mass. Linear and angular springs, k_l and k_ϕ, and dampers, c_l and c_ϕ, are attached to the shell. Again, a force $F = F_{\text{ext}} - ma_l$ and a torque $\tau = \tau_{\text{ext}} - Ja_\phi$ act on the shell. We calculate the coupled equation of motion with the Euler-Lagrange equations as

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_{l/\phi}}\right) - \left(\frac{\partial L}{\partial q_{l/\phi}}\right) = Q_{l/\phi}, \tag{4}$$

where $Q_{l/\phi} = F/\tau - c_{l/\phi}\dot{q}_{l/\phi}$ are the two generalized forces and $L = T - V$ is the Lagrangian. The kinetic energy T and the potential energy V for our 2D system are calculated as

$$\begin{aligned} T &= \frac{1}{2}m\left(\left(\cos(q_\phi)\dot{q}_\phi d + \dot{q}_l\right)^2 + \left(\sin(q_\phi)\dot{q}_\phi d\right)^2\right) + \frac{1}{2}J\dot{q}_\phi^2, \\ V &= \frac{1}{2}k_l q_l^2 + \frac{1}{2}k_\phi q_\phi^2 - mg\cos(q_\phi)d, \end{aligned} \tag{5}$$

where g is the gravitational constant. Solving Eq. (4) results in two coupled equations of motion:

$$\ddot{q}_l = f_l\left(q_l, q_\phi, \dot{q}_l, \dot{q}_\phi, F, \tau\right) \tag{6}$$

$$\ddot{q}_\phi = f_\phi\left(q_l, q_\phi, \dot{q}_l, \dot{q}_\phi, F, \tau\right) \tag{7}$$

In the following, we will denote the linear deflections along the axes of the sensor as x_l, y_l and z_l and the angular deflections as x_ϕ, y_ϕ and z_ϕ. For our sensory system, we model x_l and y_ϕ as well as y_l and x_ϕ as coupled quantities according to Eqs. (6) and (7). The origins of the x and y axes coincide with the center of mass of the shell, hence z_ϕ is not coupled with any other degree of freedom. Furthermore, we neglect coupling effects between z_l, x_ϕ and y_ϕ, as the angular deflections are expected to be very small and thus induced deflections in the z-direction are even smaller due to the second order dependency. Hence, z_l and z_ϕ can each be described by Eq. (3). We end up with a non-linear second order differential equation of the form

$$\begin{aligned} \mathbf{M}(\ddot{\mathbf{q}}) + \mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{F}) &= \mathbf{0}, \\ \text{with} \quad \mathbf{q} &= \begin{pmatrix} x_l & y_l & z_l & x_\phi & y_\phi & z_\phi \end{pmatrix}^T \\ \text{and} \quad \mathbf{F} &= \begin{pmatrix} F_x & F_y & F_z & \tau_x & \tau_y & \tau_z \end{pmatrix}^T. \end{aligned} \tag{8}$$

As the next step, we linearize Eq. (8) around $(\mathbf{q}^T, \dot{\mathbf{q}}^T, \mathbf{F}^T) = \mathbf{0}$ using the first two terms of the Taylor expansion and the linear dependency on $\ddot{\mathbf{q}}$, and we obtain

$$\bar{\mathbf{M}}\ddot{\mathbf{q}} + \bar{\mathbf{C}}\dot{\mathbf{q}} + \bar{\mathbf{K}}\mathbf{q} = \bar{\mathbf{F}}. \tag{9}$$

We can formulate this as the first order differential equation

$$\dot{\mathbf{x}}_\mathbf{q} = \underbrace{\begin{pmatrix} \mathbf{0} & \mathbf{I} \\ \bar{\mathbf{M}}^{-1}\bar{\mathbf{K}} & \bar{\mathbf{M}}^{-1}\bar{\mathbf{C}} \end{pmatrix}}_{\mathbf{A}} \mathbf{x}_\mathbf{q} + \underbrace{\begin{pmatrix} \mathbf{0} \\ \bar{\mathbf{M}}^{-1}\mathbf{I} \end{pmatrix}}_{\mathbf{B}} \bar{\mathbf{F}}, \tag{10}$$

where $\mathbf{x}_\mathbf{q} = \begin{pmatrix} \mathbf{q} & \dot{\mathbf{q}} \end{pmatrix}^T$ represents the system state with system matrix $\mathbf{A}$, input matrix $\mathbf{B}$ and identity matrix $\mathbf{I}$.

We now discretize Eq. (10) with the time step Δt to

$$\mathbf{x}_\mathbf{q}(k+1) = \mathbf{A}_\mathbf{k}\mathbf{x}_\mathbf{q}(k) + \mathbf{B}_\mathbf{k}\mathbf{F}(k) \tag{11}$$

$$\text{with} \quad \mathbf{A}_\mathbf{k} = \exp(\mathbf{A}\Delta t) \tag{12}$$

$$\text{and} \quad \mathbf{B}_\mathbf{k} = \mathbf{A}^{-1}\left(\mathbf{A}_\mathbf{k} - \mathbf{I}\right)\mathbf{B}. \tag{13}$$

$\mathbf{F}(k)$ is a partially unknown input to our system. While we have (noisy) measurements for the acceleration from our IMU, we cannot measure $\mathbf{F}_{\text{ext}}$ directly.

We assume that the force $\mathbf{F}$ does not change drastically from one time step k to the next and use the assumption

$$\mathbf{F}(k+1) \approx \mathbf{F}(k) \tag{14}$$

to restructure Eq. (11) to

$$\mathbf{x}(k+1) \approx \underbrace{\begin{pmatrix} \mathbf{A}_\mathbf{k} & \mathbf{B}_\mathbf{k} \\ \mathbf{0} & \mathbf{I} \end{pmatrix}}_{\hat{\mathbf{A}}} \mathbf{x}(k), \tag{15}$$

$$\text{with} \quad \mathbf{x} = \begin{pmatrix} \mathbf{x}_\mathbf{q} \\ \mathbf{F} \end{pmatrix} \tag{16}$$

as the new state vector.

The force-torque sensor measures the forces and torques with strain gauges. Therefore, we assume that the measured force $\mathbf{F}_\mathbf{m}$ is proportional to the deflection $\mathbf{q}$. We hence formulate the measurement equation as

$$\hat{\mathbf{F}}_\mathbf{m} = \begin{pmatrix} \mathbf{F}_\mathbf{m} \\ \mathbf{0} \end{pmatrix} = \underbrace{\begin{pmatrix} \mathbf{K} & \mathbf{0} \end{pmatrix}}_{\mathbf{C}} \mathbf{x}(k), \tag{17}$$

where $\mathbf{K}$ is a diagonal matrix, consisting of the six linear and angular spring constants k_l and k_ϕ.

$\mathbf{F}$ is now part of our state space and we use a standard Kalman filter to estimate it together with the spring deflections and their derivatives, using $\hat{\mathbf{A}}$ as the state transition and $\mathbf{C}$ as the measurement matrix. The state vector has a dimension of 18. We assume Gaussian noise in the measurement and the prediction step:

$$\mathbf{x}(k+1) = \hat{\mathbf{A}}\mathbf{x}(k) + \nu_\mathbf{R}, \tag{18}$$

$$\hat{\mathbf{F}}_\mathbf{m} = \mathbf{C}\mathbf{x}(k) + \nu_\mathbf{Q}, \tag{19}$$

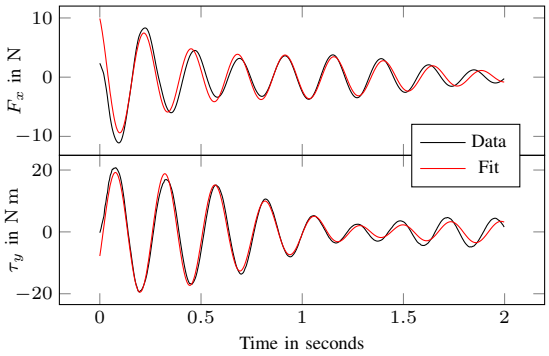

Fig. 3. The system response in x_l and y_ϕ for an external force released at $t = 0$ shows the coupling effects between both degrees of freedom.

with $\nu_\mathbf{R} \sim \mathcal{N}(0, \mathbf{R})$ and $\nu_\mathbf{Q} \sim \mathcal{N}(0, \mathbf{Q})$. The design of the measurement and prediction noise covariance matrices $\mathbf{Q}$ and $\mathbf{R}$ is crucial for the performance of the Kalman filter. For force vector $\mathbf{F_m}$ with independent components, the measurement noise

$$\mathbf{Q} = \mathrm{diag}(\sigma_\mathbf{m}) \tag{20}$$

is a diagonal matrix with standard deviations $\sigma_\mathbf{m}$.

For the prediction noise we assume that the error of the assumption in Eq. (14) introduces the dominant contribution. The control matrix $\mathbf{B_k}$ can be used to estimate how the noise $\nu_\mathbf{F}$ in $\mathbf{F}(k)$ propagates into the state space. Hence, we assume for the prediction noise

$$\nu_\mathbf{R} = \underbrace{\begin{pmatrix}\mathbf{B_k}\\ \mathbf{I}\end{pmatrix}}_{\mathbf{\Gamma}} \nu_\mathbf{F}. \tag{21}$$

The covariance of the prediction noise yields

$$\mathbf{R} = \mathrm{E}(\nu_\mathbf{R}{\nu_\mathbf{R}}^T) = \mathrm{E}(\mathbf{\Gamma}\nu_\mathbf{F}{\nu_\mathbf{F}}^T\mathbf{\Gamma}^T) = \mathbf{\Gamma}\sigma_{\mathbf{w},\mathbf{F}}\mathbf{\Gamma}^T. \tag{22}$$

We optimize for $\sigma_\mathbf{m}$ and $\sigma_{\mathbf{w},\mathbf{F}}$ by minimizing $\|\hat{\mathbf{F}}_\text{ext} - \mathbf{F}\|_1$ for the *vinyl-motion* training set introduced in Sec. V-A.

We further need to identify the model parameters of the mass-spring-damper system: the shell mass m, moments of inertia $J_{x/y}$ and J_z, the distance between mount and center of gravity d as well as the spring and damping constants $k_{l/\phi}$ and $c_{l/\phi}$. We can measure the mass directly with the force sensor when the robot is in rest and no external force is present. We calculate the moments of inertia by approximating the shell as a cylindrical barrel with negligible wall thickness of radius R and height L as

$$J_{x/y} = \frac{1}{2}mR^2 + \frac{1}{12}mL^2, \tag{23}$$

$$J_z = mR^2. \tag{24}$$

The spring and damping constants as well as the distance d are estimated from the step response of the system. To this end, we exerted constant forces by pushing on the robot shell and then introduced a step in external force by letting go of the shell. We fit the recorded system response to the harmonic solution of Eq. (8), which we solve numerically using the classical Runge-Kutta method. Fig. 3 shows the identification results for the coupled force F_x and torque τ_y.

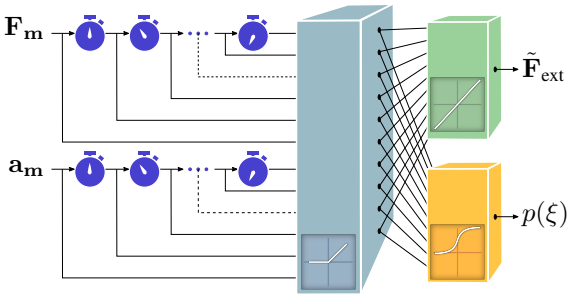

Fig. 4. Time-delay neural network. The inputs are delayed before entering the network, which splits up into one regression and one classification part.

B. Neural Network for External Force Estimation

As a second external force filter, we use a model-free approach based on a time-delay neural network (TDNN) [16]. Instead of modeling the system equations explicitly, the TDNN learns the model itself. TDNNs capture temporal information by showing the network multiple consecutive data points of a time series at one instance. To this end, the discrete input time series is delayed and buffered before entering the network. The TDNN has a feedforward structure and can be trained using standard backpropagation.

We split our problem into a regression and a classification part to exploit the superior performance of neural networks for classification tasks. Thus, our network has two network outputs as depicted in Fig. 4. The classification output $p(\xi)$ estimates whether there is currently an external force acting on the robot shell. Formally, $p(X)$ approximates the probability $\hat{p}(X)$ of the random variable $X = \{\xi, \bar{\xi}\} = \{\text{force impact}, \text{no force impact}\}$. The regression output $\tilde{\mathbf{F}}_\text{ext}$ estimates the values of the six external force components $\hat{\mathbf{F}}_\text{ext}$. Finally, we fuse both output modalities and calculate the expectancy of $\tilde{\mathbf{F}}_\text{ext}$, given $p(\xi)$ as our final estimate

$$\mathbf{F}_\text{ext} = \mathrm{E}_{p(X)}(\tilde{\mathbf{F}}_\text{ext}) \tag{25}$$

$$= \tilde{\mathbf{F}}_\text{ext} \cdot p(X = \xi) + \bar{\tilde{\mathbf{F}}}_\text{ext} \cdot p(X = \bar{\xi}), \tag{26}$$

where the external force without force impact is $\bar{\tilde{\mathbf{F}}}_\text{ext} \equiv 0$.

The input $\mathbf{x}^i = \begin{pmatrix}{\mathbf{F_m}}^{i-n:i} & {\mathbf{a_m}}^{i-n:i}\end{pmatrix}^T$ to our network consists of the measured forces and torques $\mathbf{F_m}$ and the accelerations $\mathbf{a_m}$ measured by the IMU, delayed and buffered by n time steps. The input vector is passed to the shared part of the network with two fully connected layers of 256 non-linear rectified- linear (ReLU) neurons each. The network then splits up into separated regression and classification parts of one layer with 128 neurons each. The regression part consists of neurons with linear activations, while the classification layer uses non-linear sigmoid activations.

The network parameters ϕ are optimized for all N training examples using stochastic gradient descent with momentum according to

$$\phi^* = \underset{\phi}{\mathrm{argmin}} \sum_{i=1}^{N} \mathcal{L}\left(\hat{\mathbf{y}}^i, \mathbf{y}^i\right), \tag{27}$$

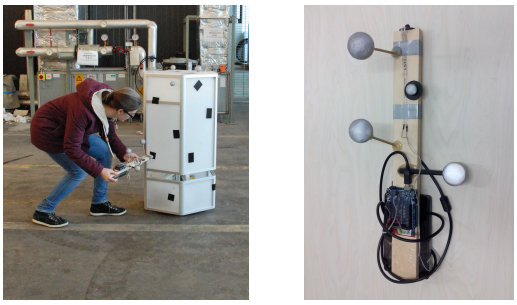

Fig. 5. Force stick used to exert and measure external forces on the robot shell. The optical markers are used to determine the 6D pose of the stick with respect to the robot.

where $\hat{\mathbf{y}}^i$ denotes the ground truth values of the estimated outputs, or labels. Our loss function $\mathcal{L}\left(\hat{\mathbf{y}}^i, \mathbf{y}^i\right)$ is a multi-task loss that comprises of the squared Euclidean norm of the regression output $\tilde{\mathbf{F}}_{\text{ext}}$ and the cross-entropy loss of the classification output $p(\xi)$:

$$\begin{aligned}\mathcal{L}\left(\hat{\mathbf{y}}^i, \mathbf{y}^i\right) &= w_r \|\tilde{\mathbf{F}}_{\text{ext}} - \hat{\mathbf{F}}_{\text{ext}}\|^2 \\ &- w_c \left[p(\xi)\log(\hat{p}(\xi)) + (1-p(\xi))\log(1-\hat{p}(\xi))\right].\end{aligned} \tag{28}$$

The weight factors w_r and w_c are hyperparameters that can be adjusted for the desired regression and classification performance. We found that $w_r = 4w_c$ yields good overall results.

V. Experiments

We performed a series of experiments to evaluate the performance of our sensory concept and the force filtering approaches. Sec. V-A presents our experimental setup. Sec. V-B introduces the lowpass filter that we use as a baseline approach. Finally, Sec. V-C compares the performance of our approaches when the robot is in rest to when the robot is in motion and evaluates them for different floor conditions.

A. Experimental Setup

We assembled a force stick depicted in Fig. 5 to generate the ground truth external force magnitudes. It is based on a flexible 1D pressure sensor attached to a wooden stick. A ceiling-mounted motion capture system provides the direction and impact point of the force. For the data collection, one experimenter repeatedly exerts forces of different extends and at different locations by pressing the force stick against the robot shell, as depicted in Fig. 5. We recorded data with the robot in motion and in rest. For the datasets with motion, a second experimenter teleoperates the robot to execute random translational and rotational motions at speeds of up to $0.75\,\mathrm{m\,s^{-1}}$ and $1.5\,\mathrm{rad\,s^{-1}}$, respectively. The exerted force magnitude ranges between 0 and $\approx 50\,\mathrm{N}$.

We collected one training, one validation and two test datasets in the motion capture area of our robot hall which has a vinyl flooring. The training, validation and test sets *vinyl-motion* were collected with a moving robot, the test set *vinyl-rest* when the robot was standing still. We further collected two test sets on different floor conditions, *stone-motion* and *carpet-motion*. Tab. I summarizes the characteristic properties of the datasets: The total number of data points and number of data points with force impact, the number of impact sequences (one sequence means exerting force at one impact location and letting go), the mean external force magnitude $\mu(F_{\text{ext}})$, and the mean translational and rotational velocities of the robot, $\mu(\|v\|)$ and $\mu(\|\omega\|)$. We recorded $137\,815$ data points at a sampling rate of $50\,\mathrm{Hz}$, corresponding to more than $45\,\mathrm{min}$.

TABLE I

Statistics of the used datasets.

	total frames	impact frames	impact seqn.	$\mu(F_{\text{ext}})$ [N]	$\mu(\|v\|)$ [$\mathrm{m\,s^{-1}}$]	$\mu(\|\omega\|)$ [$\mathrm{rad\,s^{-1}}$]
training						
vinyl-motion	42180	21889	251	17.54	0.34	0.13
validation						
vinyl-motion	12545	5774	67	15.66	0.40	0.12
testing						
vinyl-motion	26334	12912	161	15.93	0.32	0.12
vinyl-rest	32847	15566	163	14.78	0.00	0.00
stone-motion	13370	6475	95	16.43	0.35	0.13
carpet-motion	10539	4422	42	9.78	0.31	0.23

B. Baseline Lowpass Filter

Lowpass filters are standard tools for filtering high-frequency sensor noise from measurement data. A finite impulse response (FIR) lowpass filter calculates the filtered signal at time step t, $\hat{x}(t)$, by calculating the weighted mean of the past $N+1$ measurements $x(t-i)$:

$$\hat{x}(t) = \sum_{i=0}^{N} w_i x(t-i). \tag{29}$$

As a model-free baseline, we designed a lowpass filter by windowing [17], where the weights w_i are calculated as

$$w_i = \frac{\sin\left(2\pi f_t \left(i - \frac{M}{2}\right)\right)}{\pi\left(i - \frac{M}{2}\right)}. \tag{30}$$

All components of $\mathbf{F}_m$ are filtered separately with an individual lowpass filter. We keep the same filter order for all filters to ensure a common filter delay. We optimize the cutoff frequency f_t and the filter order M by a grid search on the *vinyl-motion* test set to get the best possible performance for the baseline. The cutoff frequencies for the force components are optimized together with the filter order, minimizing the mean absolute force magnitude error. The cutoff frequency for each torque component is optimized individually for its mean absolute torque error. We compensate for the filter delay during optimization by shifting the signal back by $\frac{M}{2}$ time steps. Thus, the resulting filter will be optimal within its capabilities, but the signal will be delayed.

C. Force and Impact Point Estimation

The first experiment compares our model-based and the neural filtering approach. The model-based Kalman filter estimates the combined force acting on the robot shell

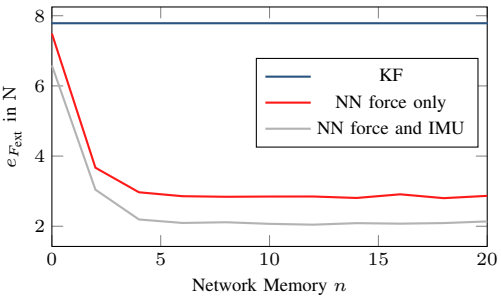

Fig. 6. Mean force magnitude error $e_{F_{\text{ext}}}$ for the Kalman filter, compared to the neural network filter for different network memories n.

and does not use acceleration measurements. Therefore, we compare it with the neural network with and without IMU information. We use the *vinyl-motion* test set and evaluate the mean absolute error between the true and estimated force magnitude. Fig. 6 shows the performance of the Kalman filter compared to the neural network filter for different network memories n. The neural network filters with and without IMU information show a smaller force magnitude error, even for $n = 0$ with only the current wrench measurements. While the network performance drops slightly without IMU information, it still clearly outperforms the Kalman filter. For the following experiments, we use the neural network filter with IMU information and a memory of 10.

The second experiment evaluates the estimation of the magnitude, direction and location of external forces on the *vinyl-rest* and *vinyl-motion* test set. Fig. 7 shows an excerpt of the ground truth and estimated external force signals with and without force filtering, with and without robot motion. Without motion, all filters resemble the true external force closely. However, the lowpass filter visibly delays the signal. When the robot is moving, the raw sensor measurements vary significantly from the true external force. The neural network filter still resembles the external force signal closely, while the performance drops visibly for both other filters. Fig. 8 compares the performance of all approaches according to the mean absolute force magnitude error $e_{F_{\text{ext}}}$, the mean distance between the true and the calculated impact point $e_{\vec{r}}$ and the mean absolute angle between the true and estimated force vector $e_{\angle F_{\text{ext}}}$. While all approaches give comparable results when the robot is in rest, the performance decreases drastically when the robot is in motion for the Kalman filter and the lowpass filter baseline. Only the neural network filter can maintain a good filtering performance.

Note that we can only evaluate the impact point distance and angular error for frames which have a ground truth impact. Furthermore, we only calculate them for predicted force magnitudes above a threshold of 5 N. All approaches meet this criterion for around 90 % of frames with ground truth impact – except the lowpass filter, which only selects approx. 80 % of frames. Additionally, a calculated impact point is only valid if it lies inside the shell geometry. For a fair comparison, we evaluate the impact point and angle error only on frames which meet the >5 N criterion and

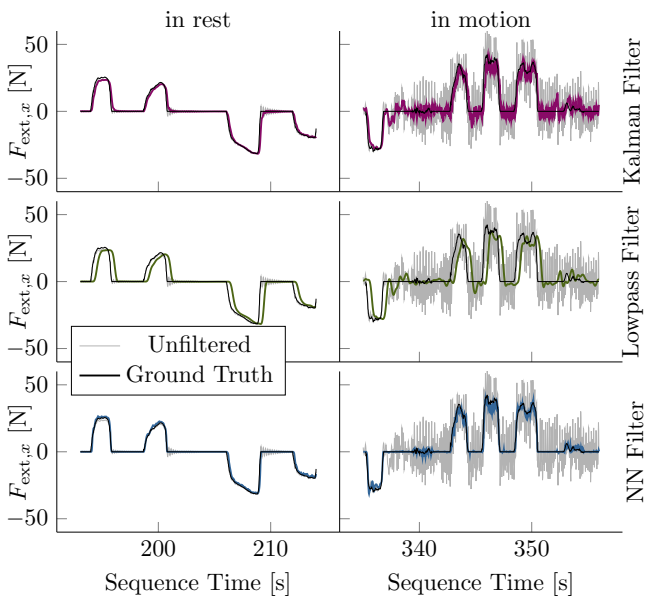

Fig. 7. Estimated and ground truth external force $F_{\text{ext},x}$ along the x-axis of the robot (pointing forward), in rest and in motion.

TABLE II

PERCENTAGE OF INVALID CALCULATED IMPACT POINTS [%]. AN IMPACT POINT IS INVALID IF IT LIES OUTSIDE THE SHELL GEOMETRY.

	Unfiltered	Lowpass Filter	Kalman Filter	NN Filter
rest	8.74	9.21	9.05	**3.42**
motion	21.87	11.91	13.22	**3.02**

are valid for all approaches. Consequently, 65.7 % of frames with ground truth impact in rest and 55.5 % in motion are evaluated for the impact point distance error. For the force angle error, 76.1 % are evaluated in rest and 77.3 % in motion. Tab. II summarizes the percentage of invalid impact points for the approaches. The neural filter outperforms all other approaches in its ability to predict a valid impact point.

The last experiment evaluates the sensitivity of our approach to varying floor conditions. The floor condition likely influences the base excitation due to robot motion, which is the main cause of force measurement errors with our setup. Tab. III presents the mean absolute force magnitude error for the *vinyl-motion*, *stone-motion* and *carpet-motion* test set. The stone and carpet datasets were not used for training the approaches. They were collected outside the motion capture area, hence we do have ground truth force directions and impact points. Interestingly, the lowpass filter and our model-based approach perform better on carpet than on the vinyl floor for which the approaches have been adjusted. Only our neural network filter performs slightly worse on carpet than on vinyl, suggesting that it implicitly learns the floor characteristics. Nevertheless, it outperforms all baselines on all tested floor conditions. The results show that our neural network filter generalizes well to previously unseen floor conditions, which confirms that our sensory concept can successfully be employed outside our robot hall.

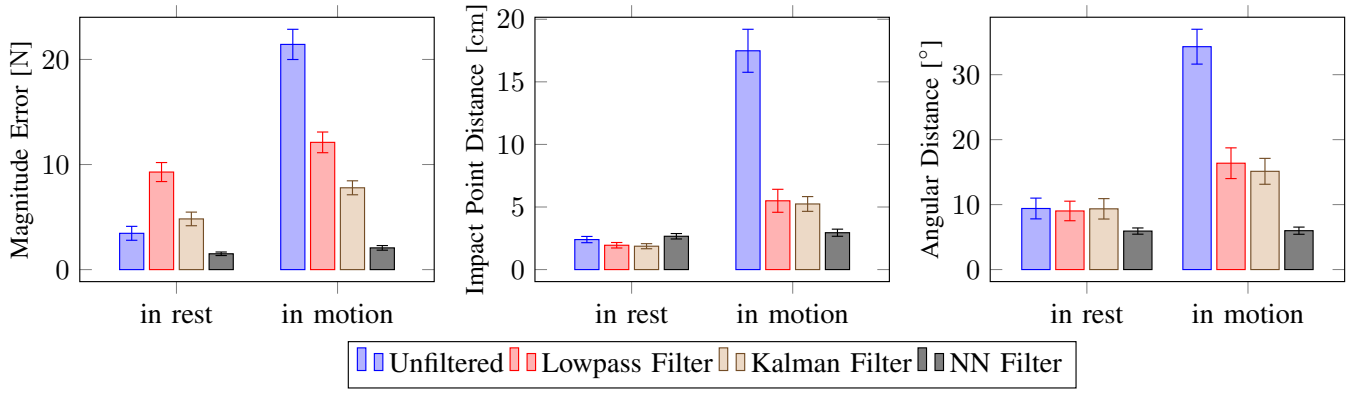

Fig. 8. Mean of the magnitude error $e_{F_{\text{ext}}}$, impact point distance $e_{\vec{r}}$ and angular error $e_{\angle \vec{F}_{\text{ext}}}$ for the unfiltered case and for the Kalman filter, lowpass filter and our neural network approach. The left columns show the errors for when the robot is in rest, the right columns for when the robot is in motion. The error bars show the standard error of the mean, amplified by a factor of 10 for visibility.

TABLE III

MEAN ABSOLUTE FORCE MAGNITUDE ERROR $\mu = e_{F_{\text{ext}}}$ AND STANDARD ERROR OF THE MEAN $SE_\mu = \sigma(e_{F_{\text{ext}}})/\sqrt{n}$, WITH SAMPLE SIZE n FOR DIFFERENT FLOOR CONDITIONS, IN N.

	Vinyl $\mu \pm SE_\mu$	Stone $\mu \pm SE_\mu$	Carpet $\mu \pm SE_\mu$
Unfiltered	21.24 ± 0.14	30.04 ± 0.24	20.00 ± 0.18
Lowpass Filter	12.52 ± 0.1	15.93 ± 0.14	8.29 ± 0.09
Kalman Filter	8.02 ± 0.06	10.12 ± 0.10	7.02 ± 0.07
NN Filter	**2.06 ± 0.02**	**2.74 ± 0.04**	**2.66 ± 0.03**

VI. CONCLUSION AND FUTURE WORK

In this paper, we presented a sensory concept for measuring interaction forces exerted onto the shell of a mobile robot. Further, we introduced a neural network-based filtering technique and a model-based baseline to distinguish external forces from those stemming from oscillations or force stimuli introduced by robot motion. Extensive experiments were carried out with our robot Canny. They demonstrate, that our tactile sensor concept enables the robot to precisely estimate the magnitude, location and direction of the impact force, even when the robot is in motion or deployed in different environments. Our neural network filtering technique clearly outperforms the model-based approach, in particular is it able to better deal with the chaotic oscillation behavior for our robot in motion. In the future, we plan to teach our robot to distinguish accidental collisions from intended interactions and investigate suitable reaction strategies for the different types of physical contact.

ACKNOWLEDGMENT

We thank Lukas Klein and Alexander Schiotka for their help during the robot experiments and Andreas Wachaja for his continuing support for all hardware and electronic issues.

REFERENCES

[1] P. Trautman and A. Krause, "Unfreezing the robot: Navigation in dense, interacting crowds," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2010.

[2] M. Kuderer, H. Kretzschmar, C. Sprunk, and W. Burgard, "Feature-based prediction of trajectories for socially compliant navigation," in *Proc. of Robotics: Science and Systems (RSS)*, 2012.

[3] E. Prassler, J. Scholz, and P. Fiorini, "A robotics wheelchair for crowded public environment," *IEEE Robotics & Automation Magazine*, vol. 8, no. 1, Feb. 2001.

[4] P. Fiorini and Z. Shiller, "Motion planning in dynamic environments using velocity obstacles," *The Int. Journal of Robotics Research*, vol. 17, no. 7, Jul. 1998.

[5] S. Thrun, M. Bennewitz, W. Burgard, A. Cremers, F. Dellaert, D. Fox, D. Hahnel, C. Rosenberg, N. Roy, J. Schulte, and et al., "Minerva: a second-generation museum tour-guide robot," *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 1999.

[6] O. Khatib, "Mobile manipulation: The robotic assistant," *Robotics and Autonomous Systems*, vol. 26, no. 2–3, Feb. 1999.

[7] S. Haddadin, A. Albu-Schäffer, and G. Hirzinger, "Requirements for safe robots: Measurements, analysis and new insights," *The Int. Journal of Robotics Research*, vol. 28, no. 11–12, Aug. 2009.

[8] Y. Hirata, T. Takagi, K. Kosuge, H. Asama, H. Kaetsu, and K. Kawabata, "Map-based control of distributed robot helpers for transporting an object in cooperation with a human," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2001.

[9] A. Sabatini, V. Genovese, and E. Pacchierotti, "A mobility aid for the support to walking and object transportation of people with motor impairments," *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2002.

[10] M. Spenko, H. Yu, and S. Dubowsky, "Robotic personal aids for mobility and monitoring for the elderly," *IEEE Trans. on Neural Systems and Rehabilitation Engineering*, vol. 14, no. 3, Sep. 2006.

[11] Y. Hirata, T. Baba, and K. Kosuge, "Motion control of omni-directional type walking support system "walking helper"," *IEEE Int. Workshop on Robot and Human Interactive Communication (ROMAN)*, 2003.

[12] L. Manuelli and R. Tedrake, "Localizing external contact using proprioceptive sensors: The contact particle filter," *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2016.

[13] K. S. Kim, T. Llado, and L. Sentis, "Full-body collision detection and reaction with omnidirectional mobile platforms: a step towards safe human-robot interaction," *Autonomous Robots*, vol. 40, no. 2, Feb. 2016.

[14] J. Frémy, F. Ferland, M. Lauria, and F. Michaud, "Force-guidance of a compliant omnidirectional non-holonomic platform," *Robotics and Autonomous Systems*, vol. 62, no. 4, Apr. 2014.

[15] A. Bicchi, J. K. Salisbury, and D. L. Brock, "Contact sensing from force measurements," *The Int. Journal of Robotics Research*, vol. 12, no. 3, Jun. 1993.

[16] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. Lang, "Phoneme recognition using time-delay neural networks," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 37, no. 3, Mar. 1989.

[17] J. Proakis and D. Manolakis, *Digital Signal Processing*. Pearson Prentice Hall, 2007.