Data Fusion for Robotic Assembly Tasks Based on Human Skills

Rui Cortesão, *Member, IEEE*, Ralf Koeppe, Urbano Nunes, *Member, IEEE*, and Gerd Hirzinger, *Fellow, IEEE*

***Abstract*—This paper describes a data fusion architecture for robotic assembly tasks based on human sensory-motor skills. These skills are transferred to the robot through geometric and dynamic perception signals. Artificial neural networks are used in the learning process. The data fusion paradigm is addressed. It consists of two independent modules for optimal fusion and filtering. Kalman techniques linked to stochastic signal evolutions are used in the fusion algorithm. Compliant motion signals obtained from vision and pose sense are fused, enhancing the task performance. Simulations and peg-in-hole experiments are reported.**

***Index Terms*—Artificial neural networks (ANNs), compliant motion signals, data fusion, Kalman filters.**

I. Introduction

The development of computer and sensor technologies allows robotic systems to have a great variety of sensors to obtain better information from the environment. Therefore, multisensor data fusion has a key role in robotic systems, addressing the problem of data combination from multiple sensors. The fusion of sensory data in a proper way improves system performance and robustness to sensor failure. The literature on data fusion is very extensive and covers a wide range of techniques [9], [13], [27]. Veeravalli and coauthors [28] introduced a Bayesian framework for a data fusion system, where each sensor sends a sequence of messages to the fusion center. Mohammad-Djafari [23] discussed classical probabilistic methods (such as maximum entropy, maximum likelihood, and Bayesian methods) to perform data fusion. Multidimensional data association in multisensor fusion applied to large-scale tracking problems is presented in [17]. Chang and coauthors [3] examined the track-to-track fusion problem (i.e., how to combine different sensor information coming from the same target), comparing different methods. New aspects of data fusion including fuzzy logic, random set theory, and conditional event algebra are addressed in [12]. The introduction of practical tools to implement data fusion providing a reverse engineering method to extract rules is presented in [16]. In [29], the fusion problem is formulated for identity verification systems (e.g., security systems) with simple classifiers. A survey of general paradigms, fusion techniques, and sensor combination for multisensor integration and fusion is done by Luo and Kay [22]. In [21], Luo presented data fusion applications, pointing out future directions, such as multilevel sensor fusion and adaptive multisensor fusion. Kalman-based fusion methods assume a known state space model of the estimated variables, as well as the statistical knowledge of system and measurement noises. Measurement fusion methods based on the Kalman filter (KF) have been widely studied and may be divided into two methods [11]: 1) state vector fusion and 2) measurement fusion. The first method uses a bank of KFs to obtain state estimates that are fused to improve the global state estimate. The second method performs first the fusion, applying then the KF to the fused state. Measurement fusion methods usually provide better results. Gan and Harris showed [11] that these two methods are equivalent if the measurement matrices are identical and the sensors have independent noise sources. There are many data fusion applications that use Kalman techniques. For example, the fusion of odometry and global positioning system (GPS) data with extended Kalman filters (EKFs) is discussed in [26]. Sasiadek and Wang [25] applied adaptive fuzzy Kalman filtering techniques to fuse position signals from GPS and inertial navigation systems. In [24], the fusion of odometry and sonar data for mobile robot navigation using EKFs and adaptive fuzzy logic is presented. Drolet and coauthors [10] proposed a bank of KFs to fuse sensory information of a remotely operated vehicle.

In this study, the fusion architecture maps geometric information into desired compliant motion signals emerged from human skills. The fusion function minimizes the noise power, and the KF estimates a noiseless (i.e., no system noise) signal described statistically.

This paper is organized as follows. The proposed data fusion architecture is discussed in Section II. Section III describes the optimal fusion function based on noise power minimization. Section IV addresses the KF to estimate variables with unknown dynamics. A model-free equation capable of following arbitrary variables is proposed. An engineering approach is used to tune the parameters of interest for a certain task. Simulations are analyzed in Section V. Section VI describes the global experimental setup and presents data fusion experiments. Pose sense and data

Manuscript received August 4, 2003; revised March 4, 2004. This paper was recommended for publication by Associate Editor G. Oriolo and Editor S. Hutchinson upon evaluation of the reviewers' comments. This work was supported in part by the Portuguese Science and Technology Foundation (FCT) Project POSI/33594/SRI/2000. This paper was presented in part at the IEEE International Conference on Intelligent Robots and Systems [8].

R. Cortesão and U. Nunes are with the Institute of Systems and Robotics, University of Coimbra, 3030 Coimbra, Portugal (e-mail: cortesao@isr.uc.pt; urbano@isr.uc.pt).

R. Koeppe and G. Hirzinger are with the Institute of Robotics and Mechatronics, German Aerospace Center (DLR-Oberpfaffenhofen), 82230 Wessling, Germany (e-mail: Ralf.Koeppe@dlr.de; Gerd.Hirzinger@dlr.de).

Digital Object Identifier 10.1109/TRO.2004.832789

from two vision cameras are fused to perform the peg-in-hole task. The main conclusions are given in Section VII.

II. Data Fusion Architecture

The transfer of skills from humans to robots has many potential applications in telerobotics, manufacturing, and assembly. Human skills properly processed and analyzed can give useful insights about human control strategies, which are a key issue in the design of intelligent control systems. A schematic representation of the data fusion architecture for human–robot skill transfer is depicted in Fig. 1. There are N sets of data sources S_i that give raw information on the evolution of a certain task. The skill transfer module maps this information into skill signals based on previous training data [19]. In our setup, the skill signals consist of compliant motion signals (i.e., force and velocity) obtained from vision and pose data. The training process can be obtained by several ways. One possible way is to use the human operator to teach the task. In this case, there is a clear skill transfer from the human to the system.

III. Optimal Fusion

The fusion function receives the skill signals x_{ik} coming from the ith mapping module $(i = 1, \ldots, N)$ performing then a global fusion. Since each x_{ik} represents the same variable, a weighted mean of x_{ik} is appropriate to fuse the data. x_{ik} is modeled as

$$x_{ik} = {}^s x_k + \eta_{ik} \tag{1}$$

where ${}^s x_k$ is the signal at time k and η_{ik} is zero-mean white noise, originated by the skill transfer module and also by sensor characteristics. Thus, the fusion function has the form

$$y_k = {}^s x_k + \frac{\sum_{i=1}^{N} l_i \eta_{ik}}{\sum_{j=1}^{N} l_j} = {}^s x_k + \eta_k \tag{2}$$

where the weights l_i have to be determined to minimize the measurement noise η_k. It can be shown [4], [5] that

$$l_r = l_1 \frac{\sigma^2_{\eta_{1k}}}{\sigma^2_{\eta_{rk}}}, \quad \text{with } r = 1, \ldots, N. \tag{3}$$

The minimum noise variance is then given by

$$\sigma^2_{\eta_k} = \frac{1}{\left(\sum_{i=1}^{N} \frac{1}{\sigma^2_{\eta_{ik}}}\right)} \tag{4}$$

that is independent of l_r. The quantification of η_{ik} is hard, since the skill transfer module is seen as a nonlinear black box for the noises coming from each S_i source. Moreover, the training process generates noise. To tackle this problem, several measurements are taken from the skill signals for a stationary situation. The variance of η_{ik} can be estimated through the variance of the measurements.

IV. Optimal Filtering

An adaptive LKF is proposed to perform the optimal estimate of the fused vector.

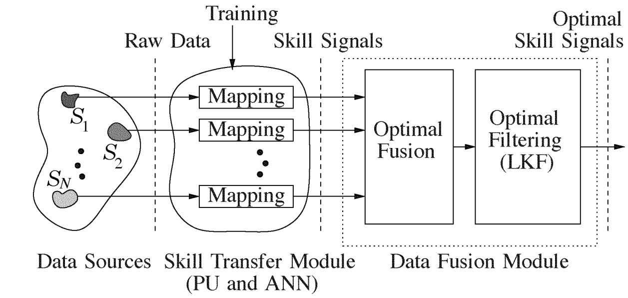

Fig. 1. Data fusion architecture. The different information coming from the S_i sources (raw data) is mapped by the skill transfer module into skill signals. Each mapping module has a preprocessing unit (PU) and an artificial neural network (ANN). The data fusion module generates an optimal vector without knowing the characteristics of the data to be fused. The optimal fusion function is based on noise power minimization and the optimal filtering uses linear KF (LKF) techniques.

A. LKF Design

1) System Model: A system equation able to track a wide range of functions may be given by

$${}^s x_k - {}^s x_{k-1} = \xi_k \tag{5}$$

where ξ_k is a zero-mean Gaussian variable that gives information about the evolution of ${}^s x_k$. Qualitatively, (5) says that the derivative of ${}^s x_k$ has a random distribution. If ${}^s x_k$ has constant shape, its derivative has zero mean, being well described by (5). However, if ${}^s x_k$ has another shape, its first derivative is not well defined by the statistical properties of ξ_k. It is necessary to go to the Nth derivative to find a good match with a zero-mean Gaussian variable.

Defining ${}^N\Omega_k$ as the Nth-order evolution of ξ_k, given as follows:

$${}^N\Omega_k = {}^{N-1}\Omega_k - {}^{N-1}\Omega_{k-1}, \quad \text{with } {}^0\Omega_k = \xi_k \tag{6}$$

the general form of (5) is

$${}^s x_k = \sum_{j=1}^{N} (-1)^{j+1} \frac{N!}{j!(N-j)!} {}^s x_{k-j} + {}^{N-1}\Omega_k. \tag{7}$$

Defining the system state as

$${}^N x_k = \begin{bmatrix} {}^s x_{k-(N-1)} & {}^s x_{k-(N-2)} & \cdots & {}^s x_{k-1} & {}^s x_k \end{bmatrix}^T \tag{8}$$

(7) can be written as

$${}^N x_k = \Phi_f \, {}^N x_{k-1} + {}^N \xi_k \tag{9}$$

with

$$\Phi_f = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ a_N & a_{N-1} & a_{N-2} & \cdots & a_1 \end{bmatrix} \tag{10}$$

$$a_i = (-1)^{i+1} \frac{N!}{i!(N-i)!}, \quad i = 1, \ldots, N \tag{11}$$

$${}^N \xi_k = \begin{bmatrix} 0 & 0 & \cdots & 0 & {}^{N-1}\Omega_k \end{bmatrix}^T. \tag{12}$$

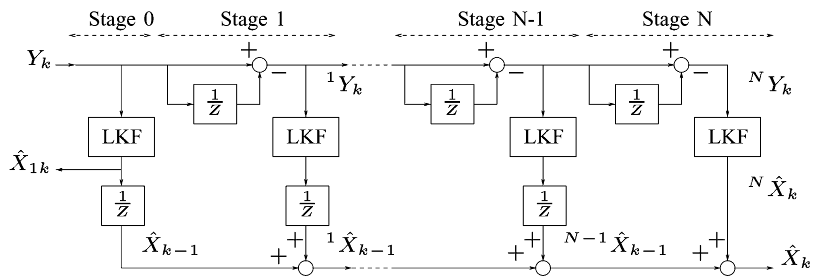

Fig. 2. Evolutionary filtering. Bank of LKFs.

2) Measurement Model: The measurement model is given by (2) that has the state space representation

$$y_k = C_f\,{}^N x_k + \eta_k \tag{13}$$

with

$$C_f = [0 \quad 0 \quad \cdots \quad 0 \quad 1]. \tag{14}$$

3) LKF Equations: Knowing (9) and (13), it is straightforward to write the LKF equations [2], [15]. The state estimate ${}^N\hat{x}_k$ is

$${}^N\hat{x}_k = \Phi_f\,{}^N\hat{x}_{k-1} + K_k\left(y_k - C_f\Phi_f\,{}^N\hat{x}_{k-1}\right) \tag{15}$$

with

$$K_k = P_{1k}C_f^T\left[C_fP_{1k}C_f^T + R_k\right]^{-1} \tag{16}$$

$$P_{1k} = \Phi_fP_{k-1}\Phi_f^T + Q_k \tag{17}$$

$$P_k = P_{1k} - K_kC_fP_{1k}. \tag{18}$$

The Q_k design may take into account online data, if

$$Q_k = E\left\{{}^N\xi_k\,{}^N\xi_k^T\right\} \tag{19}$$

does not give good results, which happens when $\sigma^2_{{}^N\xi_k{}^N\xi_k{}^T}$ is high.

B. Evolutionary Filtering

This section shows that a bank of LKFs for the optimal filtering module does not improve the fusion results. Considering a stochastic process $\{Y_k\}$ represented by

$$\{Y_k\} = \{X_k\} + \{\eta_k\} \tag{20}$$

where $\{X_k\}$ is a noiseless process and η_k is a Gaussian random variable with zero mean, it can be shown that[1]

$$Y_k = {}^NY_k + \sum_{i=1}^{N-1} {}^iY_{k-1} + Y_{k-1} \tag{21}$$

$$\sigma^2_{{}^NY_k} = \sigma^2_{{}^NX_k} + \sigma^2_{{}^N\eta_k} \tag{22}$$

$$\sigma^2_{{}^N\eta_k} = \sigma^2_{\eta_k}\sum_{i=0}^{N}\left(\frac{N!}{i!(N-i)!}\right)^2. \tag{23}$$

${}^iY_k$ and ${}^i\eta_k$ are the ith-order evolutions of Y_k and η_k, respectively,[2] i.e.,

$${}^iY_k = {}^{i-1}Y_k - {}^{i-1}Y_{k-1} \tag{24}$$

$${}^i\eta_k = {}^{i-1}\eta_k - {}^{i-1}\eta_{k-1}. \tag{25}$$

[1]See (52), (55), and (58) in the Appendix.

[2]For $i = 0$, ${}^0Y_k = Y_k$ and ${}^0\eta_k = \eta_k$.

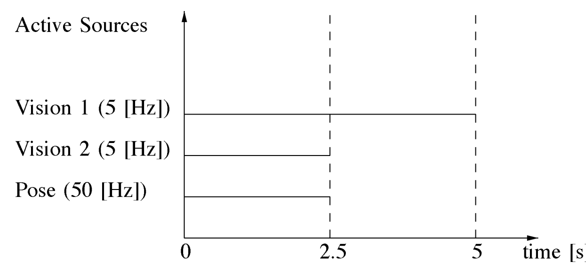

Fig. 3. Active sources.

From (20), (24), and (25)

$${}^iY_k = {}^iX_k + {}^i\eta_k \tag{26}$$

where ${}^iX_k$ is the ith-order evolution of X_k. If $\{X_k\}$ is described by

$$\{X_k\} = \{f(X_{k-j})\} + \{\xi_k\} \tag{27}$$

with $f(\cdot)$ a linear function, $j \geq 1$ and $f(0) = 0$, the same analysis of (20)–(26) can be done, i.e.,

$${}^iX_k = f\left({}^iX_{k-j}\right) + {}^i\xi_k \tag{28}$$

and the power of ${}^i\xi_k$, $\sigma^2_{{}^i\xi_k}$, changes the same way as $\sigma^2_{{}^i\eta_k}$. Hence, a LKF based on (28) and (26) has always the same Kalman gain, regardless of stage i. Then, the cascade filtering of (21) represented in Fig. 2 entails the same results as filtering only Y_k (the variables $\hat{X}_{1k}$ and $\hat{X}_k$ in Fig. 2 are equivalent).

V. Simulation Results

Simulation results obtained in the Matlab/Simulink environment are discussed in this section. The data fusion of signals with known characteristics is addressed in Section V-A. Filtering nonlinear signals with unknown characteristics is discussed in Section V-B.

A. Data Fusion

The data fusion module should be tested with synthetic data similar to the real experiments. The geometric perception sources S_i are one pose sense and two vision cameras (vision 1 and vision 2) that map for a certain task a desired force f_x (compliant motion signal) as follows:

$${}^sx_k = f_x. \tag{29}$$

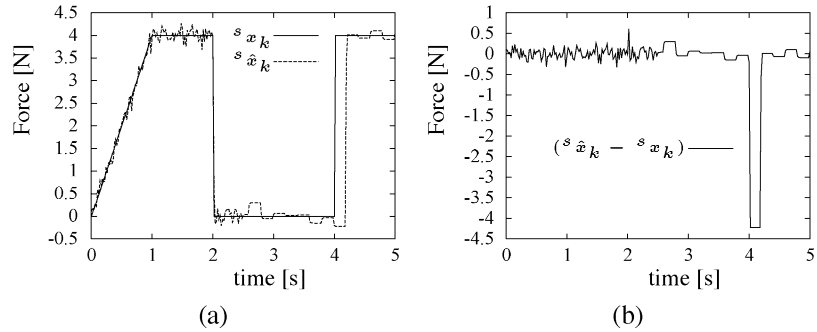

Fig. 4. Estimation of the force. (a) Desired ($^s x_k$) and estimated ($^s \hat{x}_k$) forces f_x. In the time interval [0, 2.5] s, the three sources are simultaneously active every 1/5 s ($f_s = 5$ Hz). For the other time instants ($f_s = 50$ Hz), only the pose sense is active. In the time interval [2.5, 5] s, only one vision sensor is active, giving available information every 1/50 s and updated information every 1/5 s. (b) Estimation error.

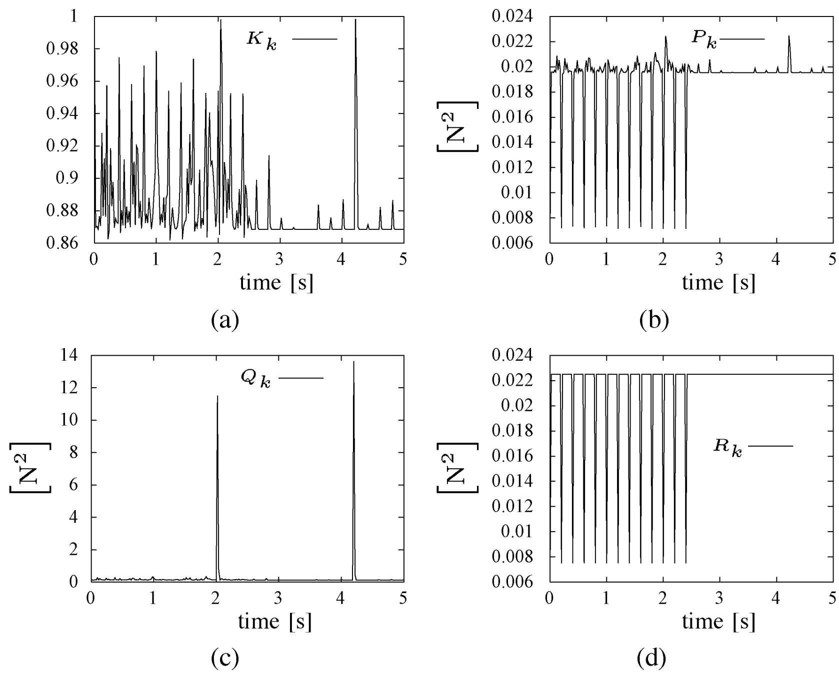

Fig. 5. Kalman matrices. (a) Kalman gain K_k. (b) Mean square error matrix P_k. (c) System evolution matrix Q_k. (d) Measurement noise matrix R_k.

The optimal filtering module uses (5) to estimate the fused variables. Fault tolerance is analyzed. Sometimes the sensors are blocked, giving no data information. Fig. 3 shows the active sources as a function of time. Vision 1 is never blocked. The pose sense and vision 2 are blocked during the time interval [2.5, 5] s. Assuming that the process $\{\xi_k\}$ is ergodic, the system evolution matrix Q_μ is

$$Q_\mu = E\left\{\left({}^s x_k - {}^s x_{k-1}\right)^2\right\} \tag{30}$$

that can be computed off-line. In the simulations, we have

$$Q_{k+1} = Q_\mu + \left({}^s \hat{x}_k - {}^s \hat{x}_{k-1}\right)^2. \tag{31}$$

The adaptive term represents the "activity" of the estimated variable $^s \hat{x}_k$ at time k. The noise referred to the output of the skill transfer module has a Gaussian distribution with variance

$$\sigma^2_{\eta_{ik}} = (0.15)^2. \tag{32}$$

The geometric sources run at different sampling times. For the vision data, a sampling frequency f_s of 5 Hz and the pose data $f_s = 50$ Hz are assumed. From (4), R_k has the value

$$R_k = \frac{(0.15)^2}{M_k} \tag{33}$$

where M_k is the number of active sensors at time k. The Kalman algorithm is given by (15)–(18), with $N = 1$, $C_f = 1$, and $\Phi_f = 1$. Q_k and R_k are, respectively, given by (31) and (33). Fig. 4 shows the estimation of f_x. The Kalman matrices are represented in Fig. 5. When the measurement error is increased due to inactive sources, K_k has smaller values [Fig. 5(a)]. If the three sources are active, the values of K_k are bigger and the mean square error of the estimate is smaller, since R_k decreases [Fig. 5(b) and (d)]. At the time instants 2.02 and 4.2 s, there is just one active source and the "activity" of $^s \hat{x}_k$ changes abruptly. K_k and P_k increase, since there is less knowledge of the process to estimate (Fig. 5(a) and (b) at 2.02 and 4.2 s). Hence, the measure of the quantity is better than the estimate based on the previous value. Fig. 5(c) shows that Q_k has almost always small values. Only during abrupt changes does Q_k increase.

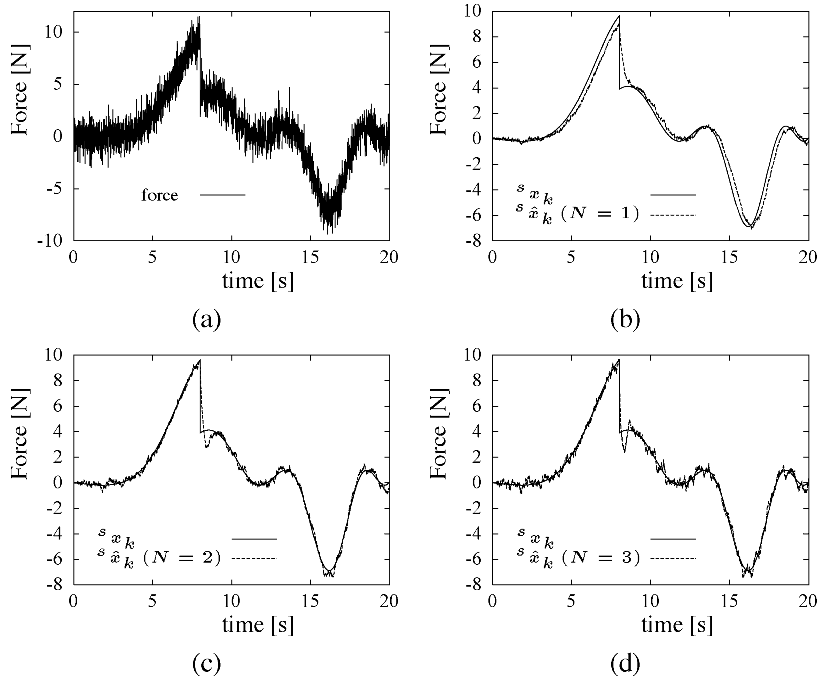

Fig. 6. Optimal filtering of nonlinear functions. (a) Signal $^s x_k$ corrupted with noise. (b) $^s x_k$ estimation with $N = 1$. (c) $^s x_k$ estimation with $N = 2$. (d) $^s x_k$ estimation with $N = 3$.

B. Optimal Filtering

This section analyzes the role of the optimal filtering module without knowing the signal characteristics. Let's consider the nonlinear function

$$^s x_k = u^3 + u^2 - u \tag{34}$$

with[3]

$$u = 2\mathrm{Chirp}\,(0.01\ [\mathrm{Hz}], 0.1\ [\mathrm{Hz}], 20\ [\mathrm{s}]) - 0.5\,\mathrm{Step}(t-8) \tag{35}$$

sampled at each 8 ms. Adding Gaussian noise to $^s x_k$ with

$$R_k = 1 \tag{36}$$

the optimal filtering performance is analyzed in the sequel. Fig. 6 illustrates simulation results for

$$Q_{^N\xi_k} = \begin{bmatrix} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma^2_{^{N-1}\Omega_k} \end{bmatrix} \tag{37}$$

with

$$\sigma^2_{^N\Omega_k} = \sigma^2_{\xi_k} \sum_{j=0}^{N} \left(\frac{N!}{j!(N-j)!} \right)^2 \cdot c^N \tag{38}$$

$$\sigma^2_{\xi_k} = \frac{R_k}{1000} \tag{39}$$

$$c = \frac{1}{500}$$

$$P_0 = 10\, Q_0$$

$$^N\hat{x}_0 = 0. \tag{40}$$

The factor c in (38) controls noise sensitivity. Fig. 6(a) shows the signal $^s x_k$ corrupted with noise. Fig. 6(b)–(d) represents the estimation of $^s x_k$, $^s \hat{x}_k$, using (7) with $N = 1$, $N = 2$, and $N = 3$, respectively. The tracking capabilities and the noise sensitivity increase with N. If $^s x_k$ does not have strong nonlinearities, (5) is enough to achieve good results $(N = 1)$.

[3]The $\mathrm{Chirp}(\cdot)$ function is a sine wave with increasing frequency.

VI. Experiments

This section reports on data fusion experiments. After the description of the robotic system, data fusion results are shown for f_x.

A. Experimental Setup

Experimental tests have been done in a robotic testbed at the German Aerospace Center (DLR-Oberpfaffenhofen). The main components of this system are as follows.

- A Manutec R2 industrial robot with a Cartesian position inner loop running at 8 ms and an input dead-time of 5 samples, equivalent to 40 ms.
- A DLR end-effector, which consists of a compliant force/torque sensor providing force/torque measurements every 8 ms. The force sensor stiffness is summarized in Table I. The manipulator compliance is lumped in the force sensor.
- A multiprocessor host computer running UNIX, enabling to compute the controller at each time step.
- Two cameras for stereo vision are mounted on the end-effector. A pneumatic gripper holds a steel peg of 30-m length and 23-mm diameter. The peg-in-hole has a clearance of 50 μm, which corresponds to a tolerance with ISO quality 9. The hole is chamferless. The environment is very stiff.

A picture of the experimental setup is depicted in Fig. 7. Fig. 7(a) represents the peg-in-hole task done by a human.

TABLE I
FORCE SENSOR STIFFNESS

Stiffness	Value
K_{w}	3 [N/mm] (x lin.)
	3 [N/mm] (y lin.)
	20 [N/mm] (z lin.)
	100 [Nm/rad] (x rot.)
	100 [Nm/rad] (y rot.)
	100 [Nm/rad] (z rot.)

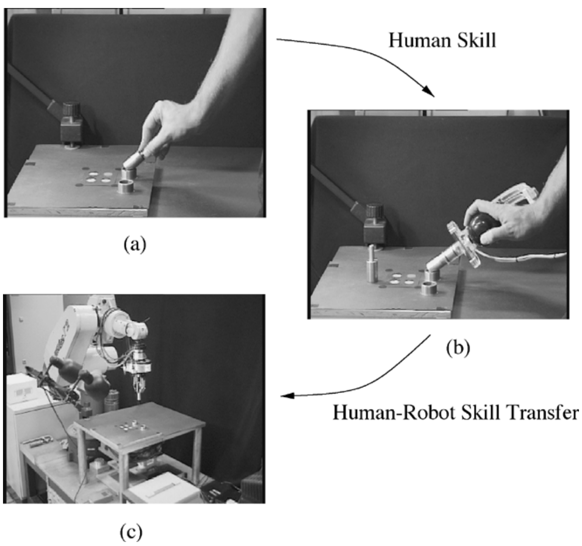

Fig. 7. Experimental setup. (a) Peg-in-hole task done by a human. (b) Peg-in-hole insertion using a teach device. Force, velocity, and pose are recorded, representing the human skill. (c) Manutec R2 robot ready to perform the peg-in-hole task after the human–robot skill transfer.

Fig. 7(b) shows the task execution with a teach device. Fig. 7(c) represents the robotic setup for the human–robot skill transfer.

B. Peg-in-Hole Insertion

The execution of the peg-in-hole task starts with the peg-in-hole with a three-point contact. The Cartesian axes and the corresponding forces and velocities are represented in Fig. 8 in peg coordinates. If the peg is perfectly aligned, the angular velocity w_y (negative sign) aligns the peg in vertical position, while f_x (negative sign) and f_z (positive sign) guarantee always a three-point contact during the alignment phase. Then, only the velocity v_z is needed to put the peg into the hole. Non-zero m_y and f_x reinforce contact with the hole inner surface during insertion. Ideally, the relevant signals for the alignment phase are f_x, f_z, and w_y. For the insertion phase, they are v_z, m_y, and f_x.

C. Peg-in-Hole Skill Transfer System

The transfer of skills from humans to robots is an important research theme and can open new perspectives to solve complex tasks. A complete mathematical characterization of human-based tasks is extremely hard, time-consuming, and very task-dependent. Starting the design from human data avoids mathematical modeling and enables one to work on the skill level, which encodes the dexterity acquired and developed

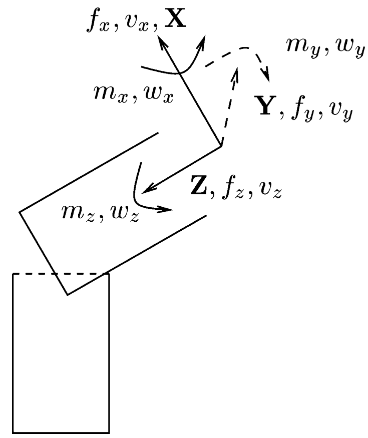

Fig. 8. Peg-in-hole with three-point contact. Cartesian axes in peg coordinates. Representation of the forces and velocities (6 DOF). w_y is the alignment velocity and v_z is the insertion velocity.

through training and experience. In the peg-in-hole task, the skill is synthesized through identification of perception-action signals from human demonstration. The skill defines an implicit representation of contact states and task trajectory, without explicit knowledge of a task model. Geometric perception signals (GEPS, i.e., signals that allow a unique classification of the task geometry) are mapped into dynamic perception signals (DYPS, i.e., signals that describe the compliant motion dynamics) based on the human skill. It is important to note that many approaches use just DYPS, which only give sparse geometric information about the contact state [20]. Similar to the human sense, GEPS correspond to vision and pose sense. The pose sense is based on robot joint angles. The relative pose can be obtained by cameras mounted on the end-effector, sensing structured or unstructured features present in the task setup. The uniqueness of the geometric information coming from sensed object features can be derived by considering the perspective n-point problem. Horaud and coauthors [14] demonstrated an analytical solution to obtain the pose of a rigid object from a single camera view for four coplanar (but not collinear) points. With stereo vision, only three coplanar points are needed to compute the pose [20]. The relative position of the coplanar points, the distance of the camera, and the focal length determine the accuracy of the visual sensing mode. In the experiments, the initial pose is reached by vision sensing. The feature pattern are four circular blobs with different colors, enabling the identification of the corresponding blobs at each time step. The color segmentation and blob tracking algorithm[4] are able to follow 2×4 blobs in 25 Hz. The compliant motion skill can be generated independently from vision or pose sense. This redundancy is fused to enhance the generation of skill maps. Fig. 9 gives an overview of the peg-in-hole skill transfer system, including the data fusion architecture and the compliant motion controller (CMC) implemented with Active Observers [6], [7]. The human skill, i.e., the human ability to perform the peg-in-hole task, is transferred to the robot by the skill transfer module trained with human data.

[4]See the work of Arbter and coauthors [1] for more details on the vision system.

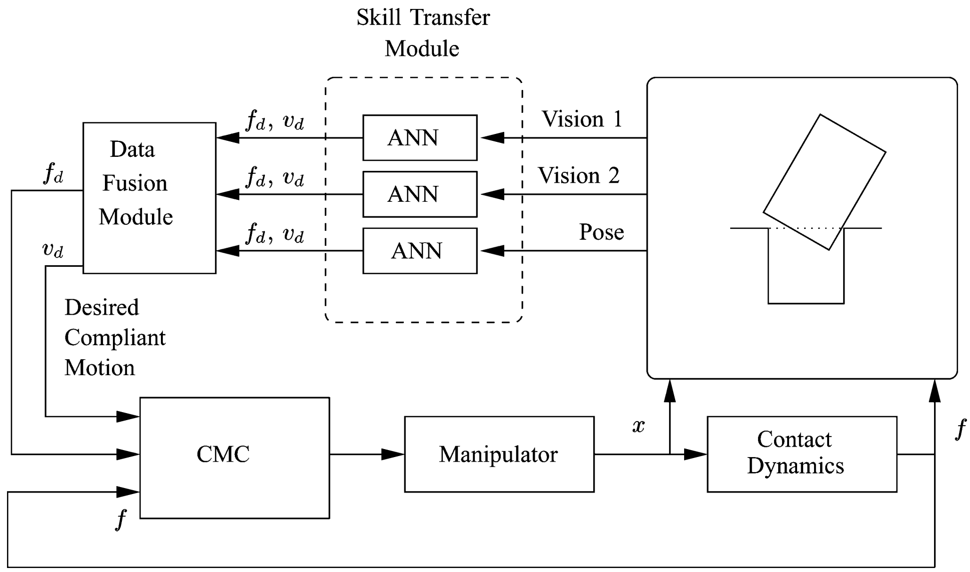

Fig. 9. Peg-in-hole skill transfer system. The data fusion module fuses the DYPS coming from the skill transfer module. The CMC controls the manipulator to achieve the desired compliant motion.

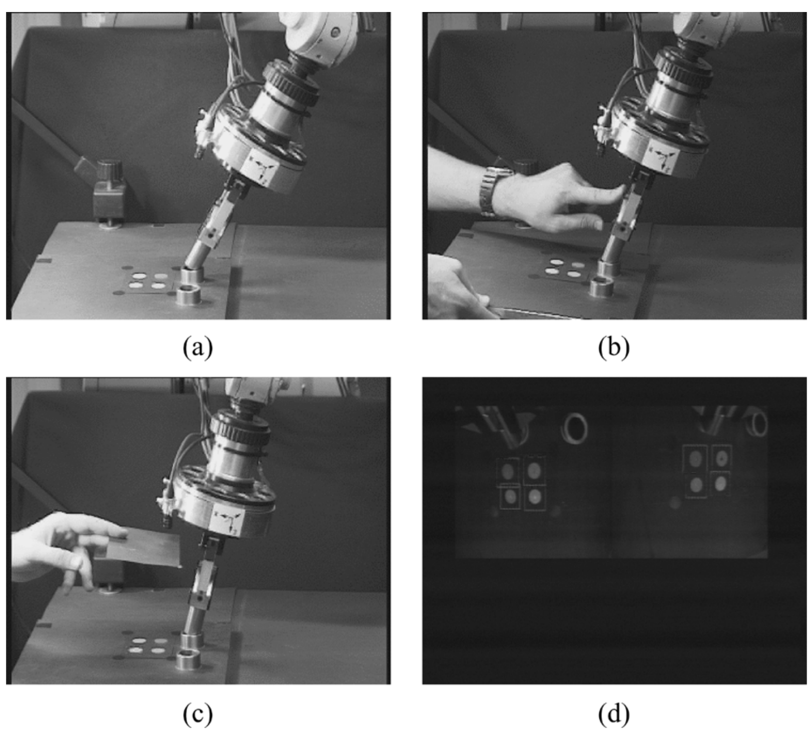

Fig. 10. Data fusion and soft robotics. (a) Peg-in-hole insertion. (b) Soft robotics. (c) Sensor failure. (d) Centroids of the four blobs.

D. Experimental Results

Fig. 10(a) illustrates the Manutec R2 robot ready to insert the peg into the hole. Since GEPS are commanding the CMC, soft robotics is possible, enabling human interference during the task execution [Fig. 10(b)]. The data fusion module takes into account the active sensors at each time step. Sensor obliteration (or sensor failure) is represented in Fig. 10(c). Fig. 10(d) shows the centroids of the four blobs captured by the vision software.

1) Artificial Neural Networks (ANNs): Two feedforward neural networks (for each camera) of size $(8 \times 10 \times 3)$ are trained to represent the skill maps of the visual sense [18]. The eight inputs represent the (x, y) position of the four blobs in pixel coordinates [Fig. 11(b) and (d)].The three outputs are f_x, f_z, and M_y for one network, and v_x, v_z, and w_y for the other. The pose sense is also trained with two feedforward neural networks of size $(3 \times 20 \times 3)$. The three inputs are the relative pose of the robot, given by two Cartesian coordinates x, z, and

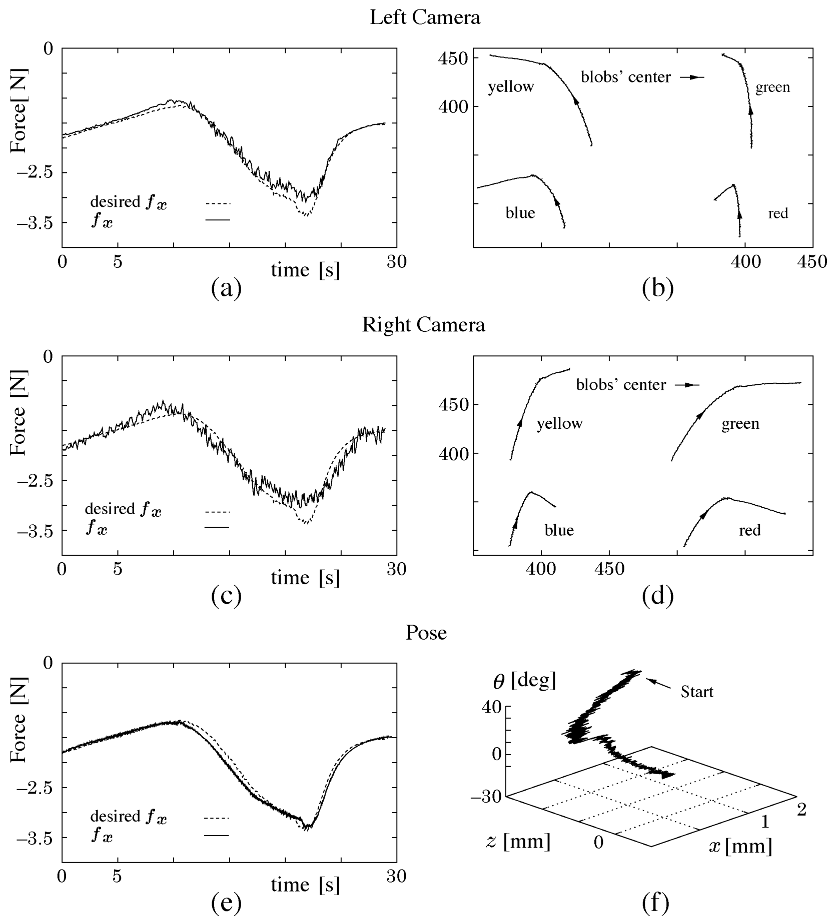

Fig. 11. ANNs for the pose and vision cameras (nominal case). (a), (c), and (e) show the output f_x for the left camera, right camera and pose, based on the GEPS represented in (b), (d), and (f), respectively.

TABLE II
f_x NOISE REFERRED TO THE OUTPUT OF THE SKILL TRANSFER MODULE

Sensor	Pose	Left Camera	Right Camera
$R_k(f_x)$	2.1×10^{-2}	9.0×10^{-1}	3.3×10^{0}

an angle θ [Fig. 11(f)]. The training phase consists of geometric shifts around the nominal case (the one used during the experiments), keeping the same compliant motion characteristics (i.e., DYPS should be invariant to small changes in GEPS). ANN weights are updated with the standard backpropagation algorithm. Pose and vision results for f_x are depicted in Fig. 11. The ANN outputs for the vision show that the right camera issues more "texture" around the nominal case than the left camera [Fig. 11(a) and (c)]. For the pose, the learned data is very close to the desired f_x [Fig. 11(e)].

2) Data Fusion: Noise analysis referred to the output of the ANNs is necessary to design the fusion module. Fig. 11(a) and (c) clearly indicate that the ANNs introduce nonstationary noise. Table II presents measurement noise for f_x and $R_k(f_x)$, taking into account not only stationary but also "learned" noise. The "learned" noise requires a trial and error tuning of R_k to smooth the learned output, without losing signal characteristics. Q_k represents the mean square evolution of the desired quantities plus an adaptive term. For f_x, the expected mean square evolution

$$Q_\mu = E\left\{(f_{x,k} - f_{x,k-1})^2\right\} \tag{41}$$

is computed off-line for the desired signal. Using (31), the full expression for Q_k is

$$Q_{k+1}(f_x) = Q_\mu + (\hat{f}_{x,k} - \hat{f}_{x,k-1})^2. \tag{42}$$

The initial conditions are $P_0 = 10Q_0$ and $\hat{x}_0 = 0$. In the experiments, the filtering module considers the system model of (5), since the compliant motion signals do not have strong nonlinearities. Figs. 12 and 13 illustrate fusion experiments.[5] Looking at Fig. 12(b) and (d) and Fig. 13(b) and (d), the alignment phase just before insertion, which occurs around 18–23 s, is faster and smoother when all sensors are active [Fig. 12(a) and (b)]. If only the vision cameras are active [Fig. 12(c) and (d)] the force response is slightly changed, since the learned functions for the vision have lower quality. Fig. 13(a) and (b) represents only the filtering process associated with the left camera. No fusion is performed (only one active sensor). Fig. 13(c) and (d) show that when a source is very noisy (right camera), the fused signal uses information mainly from the better source (pose sense). The peg-in-hole skill transfer system may be done by any geometric source (acting independently or not) with guaranteed performance, since the fused variables are very similar. Table III shows the root

[5]The measured f_x data was filtered off-line by a sixth-order low-pass Butterworth filter with a 2-Hz cut-off frequency. The phase distortion was compensated.

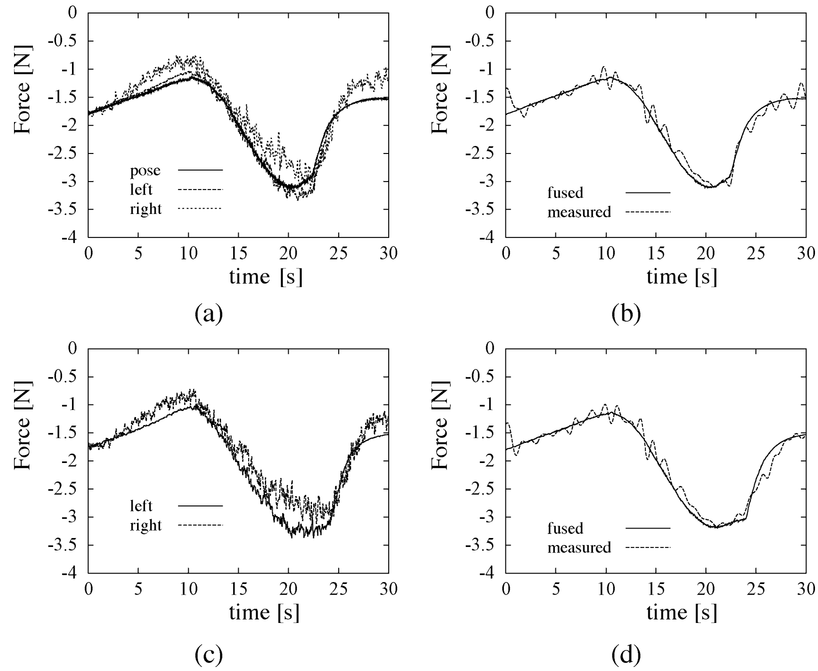

Fig. 12. Data fusion experiments for f_x. The left column represents raw data coming from the ANNs. The right column shows the fused signal versus the measured one. (a), (b) For pose sense and vision cameras active. (c), (d) These only consider vision cameras active.

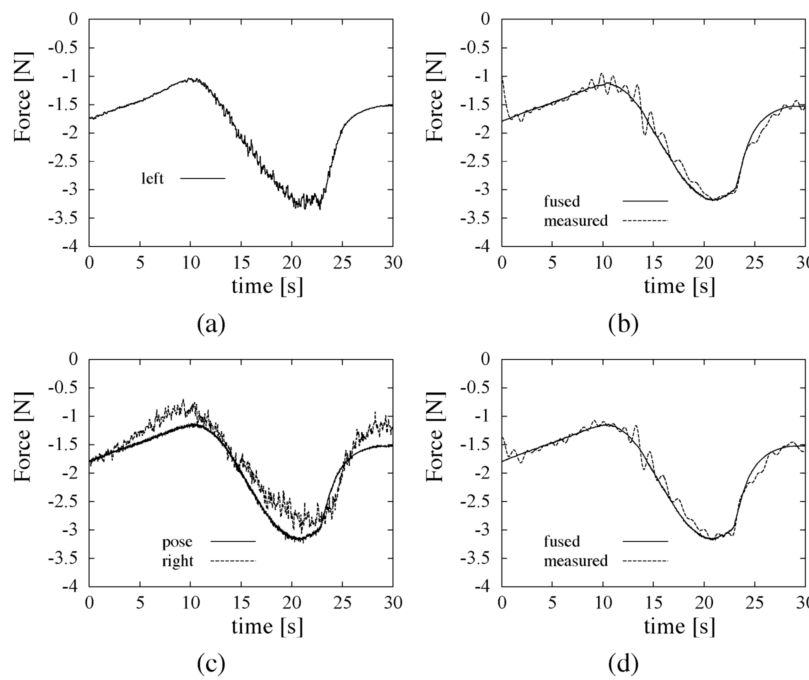

Fig. 13. Data fusion experiments for f_x. The left column represents raw data coming from the ANNs. The right column shows the fused signal versus the measured one. (a), (b) These only consider the left camera active. (c), (d) These are for right camera and pose sense active.

TABLE III
RMSE FOR THE FUSION EXPERIMENTS

Active Sensors	RMSE
Pose + Left Camera + Right Camera	0.1063
Left Camera + Right Camera	0.1894
Left Camera	0.0752
Pose + Right Camera	0.0668

mean square error (RMSE) of the fused signal f_x versus the desired force. The RMSE is small for any situation, which means that the task execution can be done by any sensory combination with similar results. The worst RMSE is for left and right cameras active, as expected, since these sensors are the noisiest ones. The RMSE for the pose and right camera active is slightly better than for the left camera alone, since the pose sense is less noisy and the associated learned function is more accurate. When all sensors are active, the RMSE is not as good as expected due to learning errors of the vision cameras [see Fig.11(a) and (c)] that, when acting together in the fused function, degrade the force estimate. It can be inferred that the overall fusion performance depends on both data fusion and skill transfer modules.

VII. Conclusion

A data fusion architecture has been presented for robotic manipulation based on human skills. It consists of a skill transfer module and a data fusion module. The skill transfer module has ANNs trained with human data that map geometric information coming from different sources into desired compliant motion signals. The data fusion module performs data fusion and Kalman filtering of the learned signals. The fusion function minimizes the noise power and the Kalman filter design is based on the stochastic description of signal evolutions. The system noise matrix describes in a stochastic way the signal evolutions in adjacent time transitions. Hence, the signals to be estimated have no associated noise. A bank of LKFs in the filtering module does not provide better results. Data fusion experiments for the peg-in-hole task have been presented, showing the importance of the fusion architecture. Vision and pose data have been fused to obtain human-like compliant motion behaviors. Fault tolerance has been analyzed. The task execution quality increases with the number and accuracy of the involved active sensors.

This data fusion architecture can be applied to other tasks with different parameters only if the available sensory information can be mapped without ambiguity into desired skill signals. ANNs can learn this mapping through sensor-based learning, affecting the overall fusion performance.

Appendix

A. Process Decomposition

A process $\{Z_n\}$ can be described by its evolutions $\{{}^iZ_n\}$. Defining $\{{}^iZ_n\}$ as

$$ {}^iZ_n = {}^{i-1}Z_n - {}^{i-1}Z_{n-1} \tag{43} $$

with

$$ {}^0Z_n = Z_n. \tag{44} $$

The Nth-order process evolution ${}^NZ_n$ may be written as

$$ {}^NZ_n = \sum_{k=0}^{N} \frac{N!}{k!(N-k)!}(-1)^k Z_{n-k}. \tag{45} $$

The mean of ${}^NZ_n$ is computed from

$$ E\left\{{}^NZ_n\right\} = \sum_{k=0}^{N} \frac{N!}{k!(N-k)!}(-1)^k E\{Z_{n-k}\}. \tag{46} $$

For a stationary process $\{Z_n\}$, its mean μ_Z is constant. From (46), we have

$$ E\left\{{}^NZ_n\right\} = \mu_Z \sum_{k=0}^{N} \frac{N!}{k!(N-k)!}(-1)^k. \tag{47} $$

Using Pascal's triangle properties, we have

$$ E\left\{{}^NZ_n\right\} = 0. \tag{48} $$

Thus, the mean of $\{{}^NZ_n\}$ is always zero if $\{Z_n\}$ is a stationary process. The ${}^NZ_n$ power is

$$ E\left\{\left({}^NZ_n\right)^2\right\} = E\left\{\left(\sum_{k=0}^{N} \frac{N!}{k!(N-k)!}(-1)^k Z_{n-k}\right)^2\right\} \tag{49} $$

that is equal to

$$ E\left\{\left({}^NZ_n\right)^2\right\} = \sum_{j=0}^{N}\sum_{k=0}^{N} \frac{N!}{k!(N-k)!}\frac{N!}{j!(N-j)!}(-1)^{(j+k)} \cdot E\{Z_{n-j}Z_{n-k}\}. \tag{50} $$

If $\{Z_n\}$ is stationary, (50) is

$$ \sigma^2_{{}^NZ_n} = \sum_{j=0}^{N}\sum_{k=0}^{N} \frac{N!}{k!(N-k)!}\frac{N!}{j!(N-j)!}(-1)^{(j+k)} R_{ZZ}(k-j). \tag{51} $$

$R_{ZZ}(k-j)$ is the autocorrelation function of $\{Z_n\}$ at the $(k-j)$ point. The ${}^NZ_n$ power strongly depends on the autocorrelation of $\{Z_n\}$.

From (43), Z_n can be written as

$$ Z_n = {}^NZ_n + \sum_{k=1}^{N-1} {}^kZ_{n-1} + Z_{n-1}. \tag{52} $$

B. Process With Uncertainty

Considering $\{Z_n\} = \{X_n\} + \{w_n\}$, where w_n is a zero-mean Gaussian variable not correlated with X_n, the mean of ${}^NZ_n$ is zero if $\{Z_n\}$ is stationary. The ${}^NZ_n$ power is

$$ E\left\{\left({}^NZ_n\right)^2\right\} = \sum_{j=0}^{N}\sum_{k=0}^{N} \frac{N!}{k!(N-k)!}\frac{N!}{j!(N-j)!}(-1)^{(j+k)} \cdot E\left\{(X_{n-j}+w_{n-j})(X_{n-k}+w_{n-k})\right\}. \tag{53} $$

Once w_n and X_n are independent, (53) can be written as

$$ \begin{aligned} E\left\{\left({}^NZ_n\right)^2\right\} = &\sum_{j=0}^{N}\sum_{k=0}^{N} \frac{N!}{k!(N-k)!}\frac{N!}{j!(N-j)!}(-1)^{(j+k)} \\ &\cdot E\left\{(X_{n-j})(X_{n-k})\right\} \\ &+ \sum_{j=0}^{N}\sum_{k=0}^{N} \frac{N!}{k!(N-k)!}\frac{N!}{j!(N-j)!}(-1)^{(j+k)} \\ &\cdot E\left\{(w_{n-j})(w_{n-k})\right\}. \end{aligned} \tag{54} $$

Using (50) and (54), we have

$$ \sigma^2_{{}^NZ_n} = \sigma^2_{{}^NX_n} + \sigma^2_{{}^Nw_n}. \tag{55} $$

Additionally, if $\{w_n\}$ is white (i.e., $\{w_n\}$ has a constant power spectral density) with power $\sigma^2_{w_n}$, the autocorrelation function of $\{w_n\}$ is the Dirac delta function, i.e.,

$$R_{ww}(n) = \sigma^2_{w_n}\delta(n). \tag{56}$$

In this case,

$$\sigma^2_{N\,w_n} = \sum_{j=0}^{N}\sum_{k=0}^{N}\frac{N!}{k!(N-k)!}\frac{N!}{j!(N-j)!}(-1)^{(j+k)}R_{ww}(k-j) \tag{57}$$

is equivalent to

$$\sigma^2_{N\,w_n} = \sigma^2_{w_n}\sum_{k=0}^{N}\left(\frac{N!}{k!(N-k)!}\right)^2. \tag{58}$$

References

[1] K. Arbter, G. Hirzinger, J. Langwald, G. Wei, and P. Wunsch, "Proven techniques for robust visual servo control," in *Robust Vision for Vision-Based Control of Motion*, M. Vincze and G. Hager, Eds. Piscataway, NJ: IEEE Press, 2000, ch. 9, pp. 109–125.

[2] S. M. Bozic, *Digital and Kalman Filtering*. London, U.K.: Edward Arnold, 1979.

[3] K. Chang, R. Saha, and Y. Bar-Shalom, "On optimal track-to-track fusion," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 33, pp. 1271–1276, Oct. 1997.

[4] R. Cortesão and R. Koeppe, "Multisensor data fusion for skill transfer systems," in *Proc. Int. Conf. Multisensor Fusion and Integration for Intelligent Systems (MFI)*, Taiwan, R.O.C., 1999, pp. 165–170.

[5] ——, "Sensor fusion for skill transfer systems," in *Proc. Int. Conf. Intelligent Robots and Systems (IROS)*, Kyongju, Korea, 1999, pp. 1014–1019.

[6] R. Cortesão, R. Koeppe, U. Nunes, and G. Hirzinger, "Explicit force control for manipulators with active observers," in *Proc. Int. Conf. Intelligent Robots and Systems (IROS)*, vol. 2, Takamatsu, Japan, 2000, pp. 1075–1080.

[7] ——, "Compliant motion control with stochastic active observers," in *Proc. Int. Conf. Intelligent Robots and Systems (IROS)*, Maui, HI, 2001, pp. 1876–1881.

[8] ——, "Data fusion for compliant motion tasks based on human skills," in *Proc. Int. Conf. Intelligent Robots and Systems (IROS)*, Lausanne, Switzerland, 2002, pp. 1529–1534.

[9] B. Dasarathy, *Decision Fusion*. Los Alamitos, CA: IEEE Comput. Soc. Press, 1994.

[10] L. Drolet, F. Michaud, and J. Côté, "Adaptable sensor fusion using multiple Kalman filters," in *Proc. Int. Conf. Intelligent Robots and Systems (IROS)*, vol. 2, Takamatsu, Japan, 2000, pp. 1434–1439.

[11] Q. Gan and C. Harris, "Comparison of two measurement fusion methods for Kalman-filter-based multisensor data fusion," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 37, pp. 273–280, Jan. 2001.

[12] I. Goodman, R. Mahler, and H. Nguyen, *Mathematics of Data Fusion*, W. Leinfellner and G. Eberlein, Eds. Norwell, MA: Kluwer, 1997, Mathematical and Statistical Methods.

[13] D. Hall and J. Linas, *Handbook of Multisensor Data Fusion*. Boca Raton, FL: CRC Press, June 2001.

[14] R. Horaud, B. Conio, O. Leboulleux, and B. Lacolle, "An analytic solution for the perspective 4-point problem," in *Proc. Int. Conf. Computer Vision and Pattern Recognition*, 1989, pp. 500–507.

[15] A. Jazwinsky, *Stochastic Processes and Filtering Theory*, R. Bellman, Ed. New York: Academic, 1970, vol. 64.

[16] N. Kasabov, D. Tuck, and M. Watts, "Implementing knowledge and data fusion in a versatile software environment for adaptive learning and decision making," in *Proc. Int. Conf. Data Fusion*, 1999, pp. 455–462.

[17] T. Kirubarajan, H. Wang, Y. Bar-Shalom, and K. Pattipati, "Efficient multisensor fusion using multidimensional data association," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 37, pp. 386–400, Apr. 2001.

[18] R. Koeppe, "Robot compliant motion based on human skill," Ph.D. dissertation, ETH Zürich, Switzerland, 2001.

[19] R. Koeppe, A. Breidenback, and G. Hirzinger, "Skill representation and acquisition of compliant motions using a teach device," in *Proc. Int. Conf. Intelligent Robots and Systems (IROS)*, vol. 2, 1996, pp. 897–904.

[20] R. Koeppe and G. Hirzinger, "Sensorimotor skill transfer of compliant motion," in *Robotics Research: 9th Int. Symp.*, J. Hollerbach and D. Koditschek, Eds., 2000, pp. 239–246.

[21] R. Luo, "Multisensor fusion and integration: approaches. applications and future research directions," in *Proc. Int. Conf. Advanced Robotics (ICAR)*, 2001, pp. 25–30.

[22] R. Luo and M. Kay, "Multisensor integration and fusion in intelligent systems," *IEEE Trans. Syst., Man, Cybern.*, vol. 19, pp. 901–931, Sept./Oct. 1989.

[23] A. Mohammad-Djafari, "Probabilistic methods for data fusion," in *Proc. Int. Workshop Maximum Entropy and Bayesian Methods*, 1997, pp. 57–69.

[24] J. Sasiadek and P. Hartana, "Sensor data fusion using Kalman filter," in *Proc. Int. Conf. Information Fusion*, vol. 2, 2000, pp. 19–25.

[25] J. Sasiadek and Q. Wang, "Sensor fusion based on Kalman filtering for autonomous robot vehicle," in *Proc. Int. Conf. Robotics and Automation (ICRA)*, 1999, pp. 2970–2975.

[26] R. Thrapp, C. Westbrook, and D. Subramanian, "Robust localization algorithms for an autonomous campus tour guide," in *Proc. Int. Conf. Robotics and Automation (ICRA)*, 2001, pp. 2065–2071.

[27] *Proc. IEEE*, vol. 85, Jan. 1997.

[28] V. Veeravalli, T. Başar, and H. Poor, "Decentralized sequential detection with a fusion center performing the sequential test," *IEEE Trans. Inform. Theory*, vol. 39, pp. 433–442, Mar. 1993.

[29] P. Verlinde, G. Maître, and E. Mayoraz, "Decision fusion using a multilinear classifier," in *Proc. Int. Conf. Multisource-Multisensor Information Fusion*, vol. 1, 1998, pp. 47–53.

Rui Cortesão (M'04) received the B.Sc. degree ("Licenciatura") in electrical engineering, the M.Sc. degree in systems and automation, and the Ph.D. degree in control and instrumentation from the University of Coimbra, Coimbra, Portugal, in 1994, 1997, and 2003, respectively.

He has been a Visiting Researcher at the German Aerospace Center (DLR) since 1998, where he has been involved with compliant motion control, data fusion, and steer-by-wire. In 2002, he was a Visiting Researcher at Stanford University, Stanford, CA, working on haptic manipulation. He became an Assistant Professor with the Electrical Engineering Department, University of Coimbra, in 2003 and has been a Researcher with the Institute for Systems and Robotics (ISR-Coimbra) since 1994. His research interests include data fusion, control, fuzzy systems, neural networks, and robotics.

Ralf Koeppe received the M.Sc. degree from Portland State University, Portland, OR, in 1989, the "Diplom-Ingenieur" from the University of Stuttgart, Stuttgart, Germany in 1992, and the Ph.D. degree from the Institute of Robotics, ETH Zürich, Zürich, Switzerland, in 2001, all in mechanical engineering.

Since 1992, he has been with the DLR Institute of Robotics and Mechatronics, Germany. He has been leading research projects in neuro-control since 1994, and directing research in high-fidelity telepresence and teleaction within the DFG Collaborative Research Center (SFB) since 1999. In 1996, he was a Visiting Researcher at Yoshikawa's Robot Laboratory, Kyoto University, Japan.

Urbano Nunes (S'90–M'95) received the "Licenciatura" and Ph.D. degrees in electrical engineering from the University of Coimbra, Coimbra, Portugal, in 1983 and 1995, respectively.

He joined the Electrical Engineering Department, University of Coimbra, in 1983, where he is currently an Assistant Professor. He is a Researcher of the Institute for Systems and Robotics (ISR), of which he is the coordinator of the Mechatronics Laboratory at the Coimbra site. His research interests are in mobile robotics, mechatronics, and intelligent vehicles.

Dr. Nunes serves on the Editorial Board of the *Journal on Machine Intelligence and Robotic Control*, and, since 1999, he has been a member of the IFAC technical committee on Cost-Effective Automation. He has served as General Co-Chair of ICAR 2003 and as a member of several program committees of international conferences.

Gerd Hirzinger (M'89–SM'95–F'97) received the Dipl.-Ing. degree and the Ph.D. degree from the Technical University of Munich, Munich, Germany, in 1969 and 1974, respectively.

In 1969 he joined the German Aerospace Center (DLR), Wessling, where he first worked on fast digital control systems. In 1976, he became head of the Automation and Robotics Laboratory of DLR, where he and his coworkers soon received several awards for innovative technology transfer from robotics research to applications. In 1991, he received a joint professorship from the Technical University of Munich. He was the prime investigator of the space robot technology experiment ROTEX, the first real robot in space, which flew onboard the shuttle Columbia in April of 1993.

Dr. Hirzinger was the recipient of several national and international awards, including the Joseph Engelberger Award in 1994 for achievements in robotic science and the Leibniz Award in 1995, the highest scientific award in Germany, and the JARA Award from the Japan Robotics Association. In 1996, he was the recipient of the Karl Heinz Beckurts Award, Germany's most important award for outstanding promotion of the partnership between science and industry.