Sensor Fusion Algorithms for Robotics: Bayesian Inference vs. Cortical Circuits

eingereichtes
ADVANCED SEMINAR
von

cand. ing. Florian Bergner

geb. am 30.09.1988
wohnhaft in:
Linprunstraße 6
80335 München
Tel.: 089 78794156

Lehrstuhl für
STEUERUNGS- und REGELUNGSTECHNIK
Technische Universität München

Univ.-Prof. Dr.-Ing./Univ. Tokio Martin Buss

Betreuer: M.Sc. Cristian Axenie
Beginn: 16.10.2013
Abgabe: 14.01.2014

Abstract

Sensor fusion is a commonly used technique to fuse information from multiple information sources in such a way that synergistic effects are used and the result gets more reliable. Sensor fusion is needed by robotic systems which have to navigate safely and autonomously in a noisy and dynamic environment. The robot's system state in the environment is usually estimated by mathematical and stochastical based fusion techniques like Bayesian Inference, Kalman Filter, etc.
Recent breakthroughs in psycho-physics, neurophysiology and bio-inspired computing have brought forth new sensor fusion techniques like Divisive Normalization. Divisive Normalization enables sensor fusion at neural network level and fulfills many neurally plausible empirical principles which conventional mathematically based fusion techniques cannot show.
This work explains and compares Bayesian Inference and Divisive Normalization and exemplarily shows their implementation on robotic systems. The advantages and disadvantages of these implementations are discussed afterwards.

Zusammenfassung

Sensor Fusion ist ein etabliertes Verfahren um Informationen verschiedener Informationsquellen so zu kombinieren, dass Synergieeffekte genutzt werden und somit zuverlässigere Schlussfolgerungen gezogen werden können. Sensor Fusion wird von Roboter-Systemen verwendet, um in einer sich ändernden Umgebung und bei verrauschten Signalen sicher und autonom navigieren zu können. Der Systemzustand eines Roboter-Systems in seiner Umgebung wird üblicherweise mathematisch-stochastisch geschätzt, wobei Fusionsverfahren wie Baysian Inference, Kalman-Filter und viele weitere angewendet werden.
In jüngster Vergangenheit hat es einige Fortschritte in der Forschung geben, vor allem in den Bereichen Innere Psychophysik, Neuro-Physiologie und Biologisch-Inspiriertes-Computing. Diese Fortschritte haben die Entwicklung neuer Fusionsverfahren angestoßen wie z.B. Divisive Normalization. Divisive Normalization ermöglicht Sensor Fusion auf der Ebene von neuronalen Netzwerken und steht im Einklang mit vielen plausiblen, biologisch-empirischen Prinzipen. Viele mathematisch basierte Fusionsverfahren hingegen erfüllen diese Prinzipen nicht oder widersprechen ihnen sogar.
Diese Arbeit erläutert und vergleicht die Fusionsverfahren Baysian Inference und Divisive Normalization und erörtert exemplarisch deren Anwendung in Roboter-Systemen. Abschließend werden die Vor- und Nachteile einer solchen Anwendung diskutiert.

Contents

Chapter 1

Introduction

1.1 Motivation

Robotic systems often have to work in dynamic environments. To work safely and autonomously in such environments robot systems have to update their system state with respect to environment. Because of noise and a dynamic environment the acquisition of information by a robotic system is always a stochastic process. So a robotic system cannot gain exact knowledge by making observations and must consequently find the best estimate for its system state.
In sensor fusion information of multiple sensors is combined and synergistic effects are used to make fusion based estimates better than single sensory based estimates. The main goal of sensor fusion is to increase the reliability of information.
Conventional fusion methods like Kalman Filter, Bayesian Inference and many more are well established and already used by robotic systems. All these methods use mathematical, stochastic and system theory based algorithms to find an optimal solution. Nevertheless it is often observed that robotic systems still show inferior performance to biological systems. At the second look this is understandable as nature had millions of years to optimize a system for a specific task. So way not taking advantage there?
Luckily much progress have been made recently in neuromorphic research and bio-inspired computing and recent breakthroughs in psycho-physical and neuro-physiological research have brought forth the development of bio-inspired fusion techniques like Divisive Normalization.
In this work a conventional mathematical stochastic approach called Bayesian Inference is explained and compared to the a new bio-inspired approach called Divisive Normalization. Implementation examples are introduced afterwards.

1.2 Abstract Fusion Model: Optimal Cue Integration

Optimal cue integration is referred in many papers [1], [2], [3], [4] and is the basis for performance comparisons between different approaches of sensor fusion. Optimal cue integration explains how different pieces of information – the so called *cues* – must be combined in order to fuse information in a stochastic optimal way. The cues are gained by observing the surrounding environment by a set of sensors.

1.3 Bayesian inference based sensor fusion

Bayesian inference [5] uses a mathematical stochastic way to perform *optimal cue integration*. This fusion technique uses prior knowledge about a system state, combines that knowledge with newly observed sensory information and finally infers a stochastic optimal belief of the new system state.
Bayesian inference needs a sensor model for handling sensory information and prior knowledge about a system state. Prior knowledge about a system state is often required to be represented by a system model. This requirement is simply enforced by a limited supply of computing power and data storage.
Bayesian inference can handle sensor fusion of multiple sensor inputs and it infers the best possible estimate by finding a stochastic optimal belief of the system state. Here is a brief outline about Bayesian inference.

Bayes' rule is the main part of Bayesian inference. However the most basic representation of Bayes' rule in equation 1.1 cannot be used directly for sensor fusion.

$$P(X|Z) = \frac{P(Z|X) \cdot P(X)}{P(Z)} \tag{1.1}$$

$$\text{posterior} = \frac{\text{likelihood} \cdot \text{prior}}{\text{evidence}} \tag{1.2}$$

In sensor fusion the system state is often time-dependent which means that the system state changes over time even when no new observation has been taken. Also the prior changes with every new observation so that Bayes' rule must be applied in a recursive manner. These two aspects lead to the fact that the prior is dependent on time and on the history of previously taken observations.

Bayesian recursion uses the posterior of the previous step as prior for the new step. If the system step is time independent equation 1.3 is valid. $\boldsymbol{z}^{t-1} = \{z_{t-1}, \ldots, z_0\}$ denotes the history of taken observations and $P(X|\boldsymbol{z}^{t-1})$ is the prior at step t.

$$P(X) = P(X|\boldsymbol{z}^{t-1}) = P(X_t|\boldsymbol{z}^{t-1}) = P(X_{t-1}|\boldsymbol{z}^{t-1}) \tag{1.3}$$

Bayesian inference with time-dependency uses Bayes' rule in the recursive form with time dependent system states X_t. The following rules are deduced with the help of [6]. The main difference to simple Bayesian recursion is that equation 1.3 is not fulfilled. This is explicitly expressed by equation 1.4. However the prior can still be deduced by using the previous posterior $P(X_{t-1}|\boldsymbol{z}^{t-1})$ and a time dependent *system model* $P(X_t|X_{t-1})$. This can be seen in equation 1.5.

$$P(X_t|\boldsymbol{z}^{t-1}) \neq P(X_{t-1}|\boldsymbol{z}^{t-1}) \tag{1.4}$$

$$P(X_t|\boldsymbol{z}^{t-1}) = \sum_{x_{t-1}} P(X_t|x_{t-1}) \cdot P(x_{t-1}|\boldsymbol{z}^{t-1}) \tag{1.5}$$

All together this leads to the final time dependent Bayes' rule for Bayesian inference which is presented by equation 1.7.

$$P(X_t|\boldsymbol{z}^t) = \frac{1}{P(z_t)} \cdot P(z_t|X_t) \cdot P(X_t|\boldsymbol{z}^{t-1}) \tag{1.6}$$

$$P(X_t|\boldsymbol{z}^t) = \frac{1}{P(z_t)} \cdot P(z_t|X_t) \cdot \sum_{x_{t-1}} P(X_t|x_{t-1}) \cdot P(x_{t-1}|\boldsymbol{z}^{t-1}) \tag{1.7}$$

1.4 Divisive normalization based sensor fusion

Divisive normalization [7], [4] is a biologically inspired sensor fusion technique found in many cortical circuits. Instead of using a mathematical stochastic approach which tries to mimic results of psychological and psychophysical studies divisive normalization tries to answer the question "How do neurons integrate sensory information?".

1.4.1 Empirical principles

Empirical principles are neural sensor fusion principles which where deduced by studies of biological sensor fusion. The two most important principles are *inverse effectiveness* and *spatial/temporal enhancement* [7].

The principle of inverse effectiveness states that small input activities of neurons with common receptive fields are enhanced (*super-additivity*) by multisensory neurons while strong input activities are degraded (*sub-additivity*).

The principle of spatial/temporal enhancement acts in the following way: If the input activities to a multisensory neuron are spatially or temporally close to each other then their activity is enhanced. These two principles have a great impact on the robustness of sensor fusion.

1.4.2 Divisive normalization

Overview The integration system consists of primary sensory neural input layers and a multisensory neural output layer (see figure 1.1). Unisensory neurons from different input layers which nevertheless have overlapping (common) *receptive fields* are connected to a multisensory neuron. In this way sensor registration is realized.

Divisive Normalization Divisive normalization (see figure 1.2) is the key of the integration process as it directly causes the integration process to realize the following very important empirical principles:

- the *spatial/temporal principle* of multisensory integration
- readjusting *neural weights* A_i with respect to *cue reliability*

Cue reliability and cue coherence are two very important aspects of sensor integration. Sensor integration has the primary goal to improve the reliability of information by fusing multiple cues in a stochastic optimal way. The individual cue reliabilities directly influence the best obtainable final cue reliability. See subsection 1.2.
Cue coherence on the other hand describes how similar input cues are and not how reliable they are. As empirical studies show cue coherence has an effect on the weights each input cue gets. With a changing cue coherence some cues get more weight and some get less weight [7]. The *mathematical neural combination rule* [7] (see eq. 1.8) helps to show that changing neural weights A_i with respect to cue coherence is contradictory to the principle of optimal cue integration.

$$R_{comb} = A_1 \cdot R_1 + A_2 \cdot R_2 + C \tag{1.8}$$

The mathematical neural combination rule states that the weighted linear sum of unimodal neural responses R_i is equal to the response R_{comb} of the multisensory neuron. If the neural integration process is implemented by using *probabilistic population codes* [8] – which use Bayesian inference – then the *neural weights* A_i are fixed to 1 and optimal cue integration is achieved.
This is clearly contradictory to the empirical principle where a changing cue coherence changes the neural weights A_i. This contradiction is solved by divisive normalization [7], [4].
Divisive normalization performs near optimal cue integration and the neural weights A_i can still adjust with respect to a changing cue coherence. The adjustments of A_i are caused by network-level computations and by the divisive normalization step. The fixed synaptic modality dominance weights d_i influence which modality gets more weight when the cue coherence deteriorates.

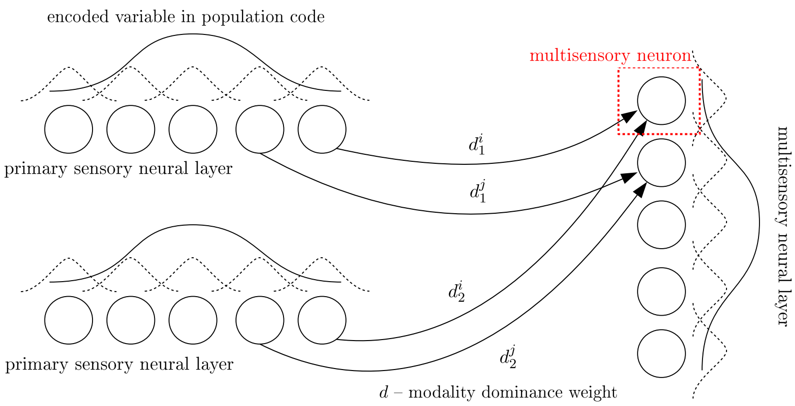
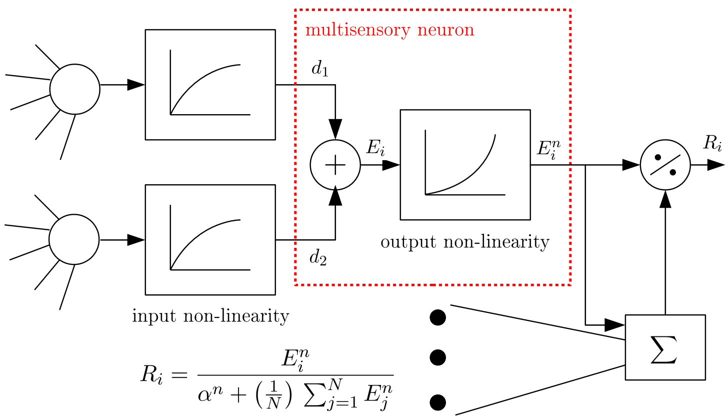


Figure 1.1: Divisive Normalization Overview



Integration algorithm

Step 1

The outputs of unisensory neurons pass an *input non-linearity* before entering the multisensory neuron. The input non-linearity is important for the suppression of consecutive stimuli within one neural input layer.

Step 2

The multisensory neuron simply combines inputs by calculating a weighted linear sum E_i of the inputs and by feeding the result to an *output non-linearity*.

Step 3

The output non-linearity is a power law with exponent n. The exponent n decides whether the empirical principle of super-additivity is applied. As long as the output non-linearity is present the principle of inverse effectiveness is always applied.

Step 4

The output of the multisensory neuron E_i^n is weighted by dividing it through the sum of the activities of all multisensory neurons of the neural output layer. This process is called *divisive normalization*.

Figure 1.2: Divisive Normalization

Chapter 2

Implementation examples

Now I want to present practical implementations for Bayesian Inference and Divisive Normalization. In this work we concentrate on the implementation of a within-modal fusion process which is used for estimating the heading of robots.

2.1 Sensor fusion based heading estimation

The estimation of heading is a common problem of technical systems like quadrocopters. But biological system cope with that problem, too.
When a robot or a biological system estimates its heading or orientation in an environment, sensory information of multiple sources must be integrated so that ambiguities are solved and system reliability increases. The integration of information can take place at multiple abstraction levels and information of multiple modalities can be fused. The implementation examples I want to present here cope with low level sensor fusion which fuses whithin-modality information taken by inertial measurements.

2.1.1 Measurement of Inertia

Technical and biological system measure inertia mostly in a similar way. Technical systems measure inertia by using a *Inertia-Measurement-Unit*, in brief *IMU*. The IMU consists of a three axis *accelerometer* and a three axis *gyroscope*. The accelerometer measures linear accelerations and the gyroscope measures angular velocities.
The biological counterpart of the IMU is called *vestibular system*. The vestibular system is located in the inner ear and consists of *semicircular canals* and *otoliths* [9], [10]. The otoliths and the semicircular canals have the same functionality as the accelerometer and the gyroscope.

2.1.2 Challenges of Heading estimation

Several challenges occur when inertial sensory information is used for estimating the heading of a system [9], [10]. These challenges can be only resolved by fusing sensory information.
The most commonly known challenges in heading estimations are the *rotation problem* [10], the *acceleration problem* [10] (also known as *gravito-intertial ambiguity* [6]) and *different regions of high accuracy* [6].

2.1.3 Theoretical frameworks for Heading Estimation

There are multiple frameworks which introduce solutions to the challenges of heading estimation. Two frameworks which are later used by the practical implementations are introduced here.

The filter framework deduces sensor fusion principles by thinking of filters [6]. See figure 2.1. The only problem of this framework is that small, constant linear accelerations cannot be determined. Biological systems have the same problem.

The observer theory states that the fusion system has an internal representation of motion variables which uses only geometrical and physical relationships. This can also be proven for biological systems [6].

2.2 The mathematical approach: Bayesian Inference

In this implementation example we have a look at how Bayesian inference can be used to estimate the heading of a robot. The implementation example bases on a model introduced by [6], [11]. This model uses Bayesian inference in combination with the observer theory framework and biological inspired sensor models.
Despite the fact that [6] introduces a complete simulation model with artificial, theoretical sensor inputs the introduced model can still be used as an estimator on robotic systems. The only change that has to be made is to feed the models with actually measured values from an IMU. In the following I briefly summarize the most important aspects of [6].

2.2.1 Introduction to the process models

Bayesian inference (see section 1.3) deduces a new estimate of the system state X by using sensory information of an observation Z. To fuse inertial observations Z

to new estimates X the following notations will be used in accordance to [6].

$$X_t = (\mathbf{\Theta}_t, \Omega_t, A_t, C_t) \tag{2.1}$$
$$Z_t = (V_t, F_t) \tag{2.2}$$

As we can see the system state X_t is a tuple of the orientation matrix $\mathbf{\Theta}_t$, the angular velocity vector Ω_t, the linear acceleration vector A_t and the so-called cupula deflection vector C_t. The cupula deflection is equivalent to angular acceleration and is calculated by a biologically inspired model of the semicircular canals [6], [12].
The observation Z_t is also a tuple. The tuple includes a vector of observed angular accelerations V_t and a vector of observed net-linear accelerations F_t.

The system model introduced by [6] uses the observer theory framework to describe physical and geometrical relationships between the motion values of the tuple X_t. The system model is needed to infer new estimates and to deduce the time dependent system propagation model $P(X_t|X_{t-1})$.

$$\mathbf{\Theta}_t = \boldsymbol{R}(\Omega_t) \cdot \mathbf{\Theta}_{t-1} \tag{2.3}$$
$$\frac{\mathrm{d}C}{\mathrm{d}t} = -\frac{1}{T_c} \cdot C - \boldsymbol{T}_{can} \cdot \frac{\mathrm{d}\Omega}{\mathrm{d}t} \tag{2.4}$$

Eq. 2.3 describes how the orientation matrix $\mathbf{\Theta}_t$ is updated. The previous rotation matrix $\mathbf{\Theta}_{t-1}$ is multiplied by a rotation matrix $\boldsymbol{R}$ while the rotation matrix itself is dependent on the current angular velocity Ω_t. The model of the semicircular canals of [12] is shown in equation 2.4. This equation introduces a relationship between angular acceleration C_t and angular velocity Ω_t. Additional information about the model can be found in [6] and [12].

The observation model of [6] also uses the observer theory. The observation model describes how inertial measurements denoted by Z are related to the motion values of X. The relationships are needed to propagate new estimations. Also the observation model has an impact on how the observation will influence the estimation over the likelihood $P(z_t|X_t)$.

$$V = C + \sigma_V \cdot \eta_t \tag{2.5}$$
$$F = \mathbf{\Theta}^{-1} (G - A) \tag{2.6}$$

As we can see in equation 2.5 additive noise can not be neglected for angular accelerations. This is only logical as additive noise causes drifts in the mathematical integration of equations 2.3 and 2.4. In equation 2.6, we can see that the gravitational acceleration G must be subtracted from the measured net-linear acceleration F. For linear accelerations the additive noise can be neglected as no integration takes place.

2.2.2 The process of recursive Bayesian Inference

The recursive Bayesian inference process uses the time-dependent recursive Bayes' rule to estimate new system states X_t and with that new motion values for the robot. As introduced by equation 1.7, Bayes' rule needs a prior $P(X_{t-1}|\boldsymbol{z}^{t-1})$, a system propagation model $P(X_t|X_{t-1})$ and a sensor model $P(z_t|X_t)$. See figure 2.2.

2.3 The bio-inspired approach: Divisive Normalization

This exemplary implementation uses a model for preprocessing inertial information in combination with divisive normalization. The preprocessing is needed as divisive normalization is a bio-inspired method which fuses information at a neural network level. The preprocessing model uses the observer theory framework and some aspects of the filter framework to calculate estimates of orientation and motion. These estimates are based on different sensory information and are represented in such a way that they can be used as inputs for divisive normalization.
The preprocessing model is based on models of [9], [10] and [6] while the concept of using divisive normalization for integrating inertial information is introduced by this work. A paper which uses divisive normalization to fuse inertial information is not known by the author.

2.3.1 Overview of the fusion process

As we can see in figure 2.3 the inertial raw data is gained by an IMU and is fed to two different integration models which exchange information.

The model of orientation calculates a gyroscope based estimation of orientation $\boldsymbol{\Theta}_t^g$ and uses divisive normalization to fuse it with an accelerometer based estimation of orientation $\boldsymbol{\Theta}_t^a$. The fusion step must fuse the two inputs according to the principles as stated by the filter framework.

The model of acceleration calculates an accelerometer based estimation of linear acceleration A_t^a (see eq. 2.7) and a gyroscope based estimation of linear acceleration A_t^g (see eq. 2.8). Then the fusion step for the final estimation of linear acceleration $\hat{A}_t$ is performed. This fusion step must again realize the principles of the filter framework. Note that A_t^a is calculated by using a high-pass filter in accordance with the filter framework principles and that all estimated linear accelerations are robot (subject) frame based. These accelerations can be transformed back into a world frame by using the estimated orientation $\hat{\boldsymbol{\Theta}}_t$.
Equation 2.9 uses $\hat{A}_t$ to calculate the accelerometer based estimation of orientation

$\mathbf{\Theta}_t^a$.

$$
\begin{aligned}
A_t^a &= f_H(F_t) && && (2.7)\\
A_t^g &= f_I^1(\mathbf{\Theta}_t^g, F_t) && \text{with} \quad f_I^1 : A_t^g = (\mathbf{\Theta}_t^g)^{-1}\ G - F_t && (2.8)\\
\mathbf{\Theta}_t^a &= f_I^2(\hat{A}_t, F_t) && \text{with} \quad f_I^2 : \mathbf{\Theta}_t^a\ \hat{A}_t = G - \mathbf{\Theta}_t^a \cdot F_t && (2.9)
\end{aligned}
$$

2.3.2 Fusing information using Divisive Normalization

The two divisive normalization based fusion steps – one for the fusion of orientations and one for the fusion of linear accelerations – can be implemented using identical principles. So here we only have an exemplary look at the fusion of orientations (see figure 2.4).
In order to fuse information by using divisive normalization values must be represented by population codes. Here $\boldsymbol{\Theta_t^g}$ and $\boldsymbol{\Theta_t^a}$ must be represented by two population codes (PCs). The activity pattern of these PCs – the activity pattern refers to the reliability of the value the pattern represents – must be adjusted. This can be done using a *reliability model* which uses a history of sensor measurements or a priory knowledge. Note that the reliability model must also act in accordance with the filter framework principles.
The outputs of the neurons of the population codes are connected to multisensory neurons of the multisensory neural layer. The neurons connected to a multisensory neuron must have common receptive fields and the modality dominance weights d_1^i and d_2^i must be chosen in such a way that the dominance of an input changes with coherence in the right manner. For example with decreasing coherence the orientation $\mathbf{\Theta}_t^a$ which is based on accelerometer measurements should get more weight so that drifts can be countered. The neurons of the multisensory neural layer form again a population code which can then be used to get the fused orientation result $\boldsymbol{\hat{\Theta}_t}$.

2.4 Conclusions

Bayesian Inference based sensor fusion insures that information is fused in the best possible way as it fuses signals in a stochastic optimal way. An implementation using Bayesian Inference – as introduced by [6] – can even show a better performance than the most commonly used Kalman Filter. This is reasonable as the introduced system is non-linear and Kalman Filters show best performance only when used with linear system models.
The introduced implementation of Bayesian Inference uses a Bayesian recursive process which cannot be implemented easily because the process uses non-linear equations which aren't solvable in an analytic way. As the system state is high-dimensional using a discrete state space is also not feasible. Therefore – as proposed by [6] – the computationally expensive method of *particle filtering* has to be used.

Performance evaluations of [6] show that this implementation of Bayesian Inference has a comparable behavior to the biological vestibular integration process. However Bayesian Inference has no explicit advantage over other technical fusion techniques. For example an Extended Kalman Filter might have the same performance.
The introduced implementation of Bayesian Inference is not very robust and multiple parameters have to be tuned. Though using some biological inspired abstract principles this implementation is realized in a mathematical stochastic way and thus have disadvantages of mathematical exact systems.

Divisive Normalization In contrast to Bayesian inference, an implementation which uses Divisive Normalization is much closer to biology and promises many improvements to traditional fusion techniques. Divisive Normalization realizes many empirical principles of neural level fusion which have been established by neurophysiology. This not only helps to make the fusion behavior more nature-like, this might also help to make fusion implementations on robots more robust.
An implementation which bases on Divisive Normalization can readjust itself to changing cue reliabilities and thus the fusion process performs near optimal cue integration. Divisive Normalization only uses simple math while the complexity of sensor fusion is handled mainly by (artificial) neural networks. Consequently Divisive Normalization is computationally less expensive.
However the introduced implementation using Divisive Normalization needs an internal motion model for preprocessing sensory information. But this model only uses simple physical relationships between motion variables and simple geometry so the model is not computationally expensive. Note that biology also implements an internal motion model [9], [10] so the need of an internal motion model can be justified by nature.
Divisive normalization has one more big advantage over Bayesian Inference and other conventional fusion techniques. In Divisive normalization cue coherence decides which sensory input gets more weight. This conflicts clearly with traditional stochastic-mathematical fusion techniques like Bayesian Inference where every sensory input gets equal weight and where thus the principle of optimal cue integration is strictly fulfilled at all times. So Divisive Normalization can only reach near optimal cue integration but adjusting the weights of sensory inputs with respect to cue coherence makes it more robust.
Despite the many improvements Divisive Normalization introduces for integrating sensory information on robots it must be noted that the performance of Divisive Normalization must still be verified in real world applications.

sensor	**sensory information**	**deduced information**
accelerometer	high frequency components of *net-linear acceleration* [10]	linear acceleration
accelerometer	low frequency components of *net-linear acceleration* [10]	orientation
gyroscope	high frequency components of rotations	orientation, rotation
accelerometer or gyroscope	fast rotations	linear accelerations are ignored
accelerometer or gyroscope		small, constant linear accelerations cannot be detected

Figure 2.1: The filter framework

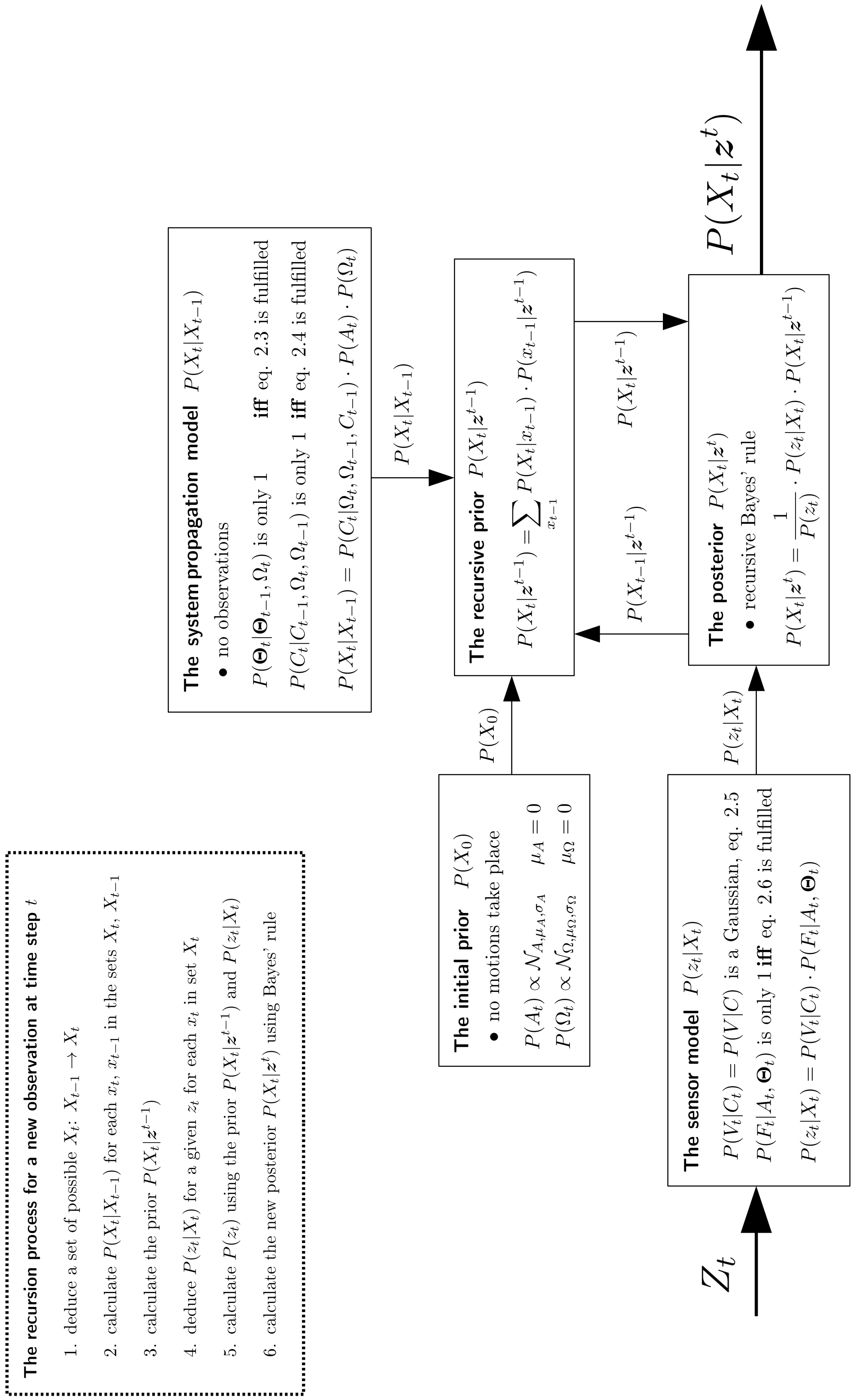


Figure 2.2: Bayesian inference process

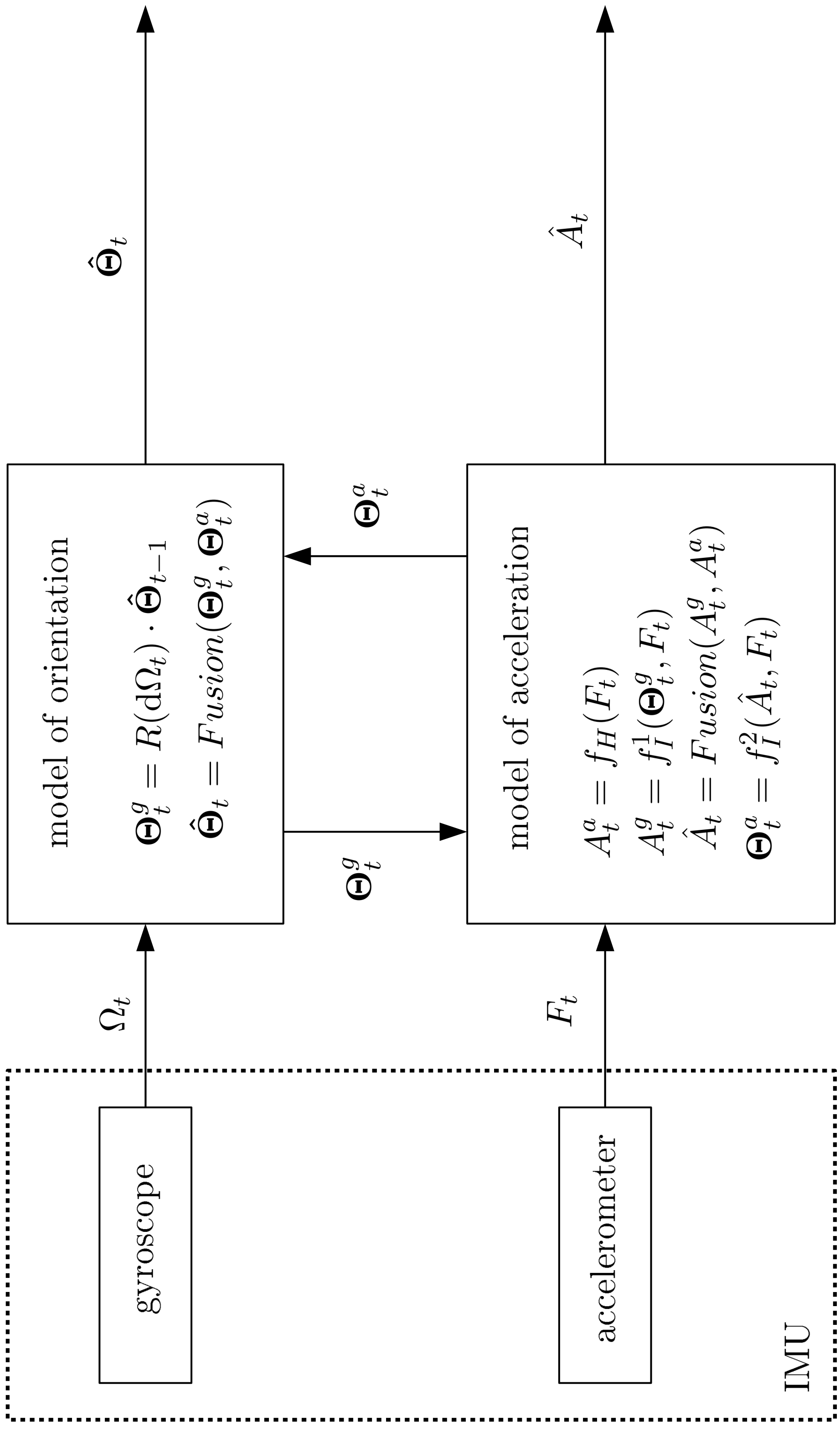


Figure 2.3: Integration Model

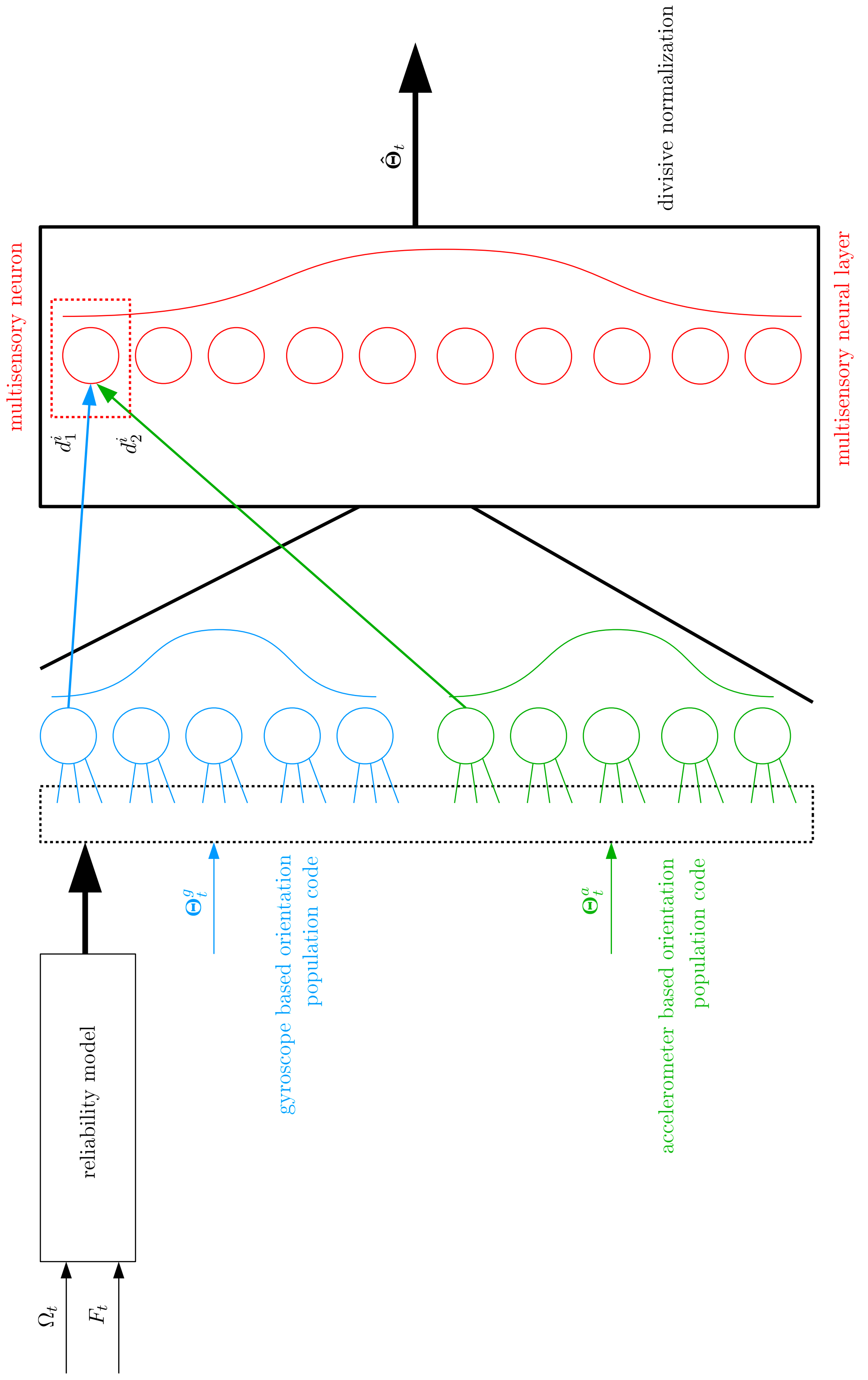


Figure 2.4: Divisive Normalization Based Fusion of Orientation

List of Figures

Bibliography

[1] Ernst, Marc O.: Optimal multisensory integration: Assumptions and limits. In Stein, Barry E. (editor): The new Handbook of Multisensory Processing, chapter 30, page p. 509 ff. The MIT Press, 2012.

[2] Ernst, Marc O. and Heinrich H. Buelthoff: Mergin the senses into a robust percept. Trends in Cognitive Sciences, vol. 8 (no. 4): pp. 163–169, April 2004.

[3] Agelaki, Dora E., Young Gu, and Gregory C. DeAngelis: Multisensory integration: psychophysics, neurophysiology, and computation. Current Opinion in Neurobiology, vol. 19 : pp. 452–448, August 2009.

[4] Fetsch, Christopher R., Gregory C. DeAngelis, and Dora E. Angelaki: Bridging the gap between theories of sensory cue integration and the physiology of multisensory neurons. Nature Reviews, Neuroscience, vol. 14 : pp. 429–442, June 2013.

[5] Durrant-Whyte, Hugh and Thomas C. Henderson: Multisensor data fusion. In Sciliano, Bruno and Oussama Khatib (editors): Springer Handbook of Robotics, chapter 25, pages pp. 585–612. Springer-Verlag Berlin Heidelberg, 2008.

[6] Jaurens, Jean and Jacques Droulez: Bayesian processing of vestibular information. Biological Cybernetics, vol. 96 : pp. 389–404, April 2007.

[7] Tomokazu Ohshiro, Dora E. Agelaki and Gregory C. DeAngelis: A normalization model of multisensory integration. Nature Neuroscience, vol. 14 (no. 16): pp. 775–782, June 2011.

[8] Ma, Wei Ji, Jeffrey M. Beck, Peter E. Latham, and Alexandre Pouget: Bayesian inference with probabilistic population codes. Nature Neuroscience, vol. 9 (no. 11): pp. 1432–1438, November 2006.

[9] Dora E. Angelaki, Aasef G. Shaikh, Andrea M. Green and J. David Dickman: Neurons compute internal models of physical laws of motion. Nature, vol. 430 : pp. 560–563, July 2004.

[10] Angelaki, Dora E. and Kathleen E. Cullen: Vestibular system: The many facets of a multimodal sense. The Annual Review of Neuroscience, vol. 31 : pp. 125–150, 2008.

[11] Jaurens, Jean, Dominik Straumann, and Bernhard J. M. Hess: Processing of angular motion and gravity information through an internal model. Journal of Neurophysiology, vol. 104 (no. 3): pp. 1370–1381, September 2010.

[12] Raphan, Theodore and Bernard Cohen: The vestibulo-ocular reflex in three dimensions. Experimantal Brain Research, vol. 145 : pp. 1–27, July 2002.

[13] Conradt, J., Simon, P., Pescatore, M., and Verschure, PFMJ. (2002). Saliency Maps Operating on Stereo Images Detect Landmarks and their Distance, Int. Conference on Artificial Neural Networks (ICANN2002), p. 795-800, Madrid, Spain.

[14] Axenie, C., Conradt, J. (2014) Cortically inspired sensor fusion network for mobile robot egomotion estimation. Robotics and Autonomous Systems, Special Issue on "Emerging Spatial Competences: From Machine Perception to Sensorimotor Intelligence", Volume 71, pages 69-82.