Generative models for action generation and action understanding

JUDITH BÜTEPAGE

Doctoral Thesis
Stockholm, Sweden 2019

KTH Royal Institute of Technology
School of Electrical Engineering and Computer Science
SE-100 44 Stockholm
SWEDEN

TRITA-EECS-AVL-2019:60
ISBN 978-91-7873-246-3

Tryck: Universitetsservice US AB

Abstract

The question of how to build intelligent machines raises the question of how to represent the world to enable intelligent behavior. In nature, this representation relies on the interplay between an organism's sensory input and motor input. Action-perception loops allow many complex behaviors to arise naturally. In this work, we take these sensorimotor contingencies as an inspiration to build robot systems that can autonomously interact with their environment and with humans. The goal is to pave the way for robot systems that can learn motor control in an unsupervised fashion and relate their own sensorimotor experience to observed human actions. By combining action generation and action understanding we hope to facilitate smooth and intuitive interaction between robots and humans in shared work spaces.

To model robot sensorimotor contingencies and human behavior we employ generative models. Since generative models represent a joint distribution over relevant variables, they are flexible enough to cover the range of tasks that we are tackling here. Generative models can represent variables that originate from multiple modalities, model temporal dynamics, incorporate latent variables and represent uncertainty over any variable - all of which are features required to model sensorimotor contingencies. By using generative models, we can predict the temporal development of the variables in the future, which is important for intelligent action selection.

We present two lines of work. Firstly, we will focus on unsupervised learning of motor control with help of sensorimotor contingencies. Based on Gaussian Process forward models we demonstrate how the robot can execute goal-directed actions with the help of planning techniques or reinforcement learning. Secondly, we present a number of approaches to model human activity, ranging from pure unsupervised motion prediction to including semantic action and affordance labels. Here we employ deep generative models, namely Variational Autoencoders, to model the 3D skeletal pose of humans over time and, if required, include semantic information. These two lines of work are then combined to implement physical human-robot interaction tasks.

Our experiments focus on real-time applications, both when it comes to robot experiments and human activity modeling. Since many real-world scenarios do not have access to high-end sensors, we require our models to cope with uncertainty. Additional requirements are data-efficient learning, because of the wear and tear of the robot and human involvement, online employability and operation under safety and compliance constraints. We demonstrate how generative models of sensorimotor contingencies can handle these requirements in our experiments satisfyingly.

Sammanfattning

Frågan om hur man bygger intelligenta maskiner väcker frågan om hur man kan representera världen för att möjliggöra intelligent beteende. I naturen bygger en sådan representation på samspelet mellan en organisms sensoriska intryck och handlingar. Kopplingar mellan sinnesintryck och handlingar gör att många komplexa beteenden kan uppstå naturligt. I detta arbete tar vi dessa sensorimotoriska kopplingar som en inspiration för att bygga robotarsystem som autonomt kan interagera med sin miljö och med människor. Målet är att bana väg för robotarsystem som självständiga kan lära sig att kontrollera sina rörelser och relatera sina egen sensorimotoriska upplevelser till observerade mänskliga handlingar. Genom att relatera robotens rörelser och förståelsen av mänskliga handlingar, hoppas vi kunna underlätta smidig och intuitiv interaktion mellan robotar och människor.

För att modellera robotens sensimotoriska kopplingar och mänskligt beteende använder vi generativa modeller. Eftersom generativa modeller representerar en multivariat fördelning över relevanta variabler, är de tillräckligt flexibla för att uppfylla dem krav som vi ställer här. Generativa modeller kan representera variabler från olika modaliteter, modellera temporala dynamiska system, modellera latenta variabler och representera variablers varians - alla dessa egenskaper är nödvändiga för att modellera sensorimotoriska kopplingar. Genom att använda generativa modeller kan vi förutse utvecklingen av variablerna i framtiden, vilket är viktigt för att ta intelligenta beslut.

Vi presenterar arbete som går i två riktningar. För det första kommer vi att fokusera på självständig inlärande av rörelse kontroll med hjälp av sensorimotoriska kopplingar. Baserat på Gaussian Process forward modeller visar vi hur roboten kan röra på sig mot ett mål med hjälp av planeringstekniker eller förstärkningslärande. För det andra presenterar vi ett antal tillvägagångssätt för att modellera mänsklig aktivitet, allt från att förutse hur människan kommer röra på sig till att inkludera semantisk information. Här använder vi djupa generativa modeller, nämligen Variational Autoencoders, för att modellera 3D-skelettpositionen av människor över tid och, om så krävs, inkludera semantisk information. Dessa två ideer kombineras sedan för att hjälpa roboten att interagera med människan.

Våra experiment fokuserar på realtidsscenarion, både när det gäller robot experiment och mänsklig aktivitet modellering. Eftersom många verkliga scenarier inte har tillgång till avancerade sensorer, kräver vi att våra modeller hanterar osäkerhet. Ytterligare krav är maskininlärningsmodeller som inte behöver mycket data, att systems fungerar i realtid och under säkerhetskrav. Vi visar hur generativa modeller av sensorimotoriska kopplingar kan hantera dessa krav i våra experiment tillfredsställande.

Dedicated to my parents

Without them I would be nowhere close to where I am now

Acknowledgements

During the last four years I had the pleasure to meet many interesting people that I learned a tremendous amount from and that I spent wonderful times with. First and foremost, I would like to thank my supervisors Danica Kragic and Hedvig Kjellström for not only guiding my development as an independent researcher but also for being role models of strong women. I would also like to thank Mårten Björkman for contributing to my work.

Secondly, I would like to thank all my office mates that I have had over the years, especially Cheng, Püren and Alessandro, who made the beginning of my PhD very easy, and Petra, who accompanied me through the end. 715 has always been an awesome office. Cheng deserves a second thank (or even more) for all those opportunities I got through her helping hands and all those interesting discussions we have had. She calls herself a part-time Bayesian - together we make one full-time employee.

I have been fortunate enough to make many friends both within and outside of RPL / CVAP. I would like to thank those friends that I made during my first and only Master's year and who still make my life more enjoyable: Thomai, Magnus and Sebastian. I would like to thank Diogo for many enjoyable conversations, Joshua for the fantastic hikes, Joao for each and every glass of Bundaberg and Michael for teaching me how to enjoy Whiskey.

Special thanks goes to Freddy, who has been my faithful companion for 23 years, and my sister Greta who I had the pleasure to share a life with here in Stockholm. Finally, I would like to thank my parents, who taught me that I can achieve almost anything if I only work hard.

Judith Bütepage
Stockholm, Sweden
September, 2019

Contents

Part I

Introduction

Chapter 1

Introduction

From its beginning, the field of artificial intelligence (AI) has been inspired by research in neuroscience and psychology [1]. While early work in the 1950ies mostly focused on computational models, such as neural networks, later ideas encompassed symbolic and logical reasoning in well-defined state and action spaces. However, it proved difficult to solve realistic problems, such as vision or natural language, with these approaches. The complexity of the feature space as well as the combinatorial explosion of required computations rendered hand-crafted state-spaces and logic reasoning infeasible.

In the second half of the 1980, the concept of embodied intelligence revolutionized artificial intelligence [2]. Instead of logical architectures and knowledge representation, the embodied view argues that intelligent behavior emerges naturally from the interplay between motor and sensory channels [3]. This coupling between an agent and its environment through sensorimotor signals and constant inference and feedback loops is suggested to account for complex behavior without the need of high-level reasoning. In this view, internal symbolic representations become obsolete because the environment is its own representation that is actively sampled via sensory channels and manipulated by self-induced actions.

As an example, consider an agent that is asked to interact with indoor environments such as depicted in Figure 1.1 a). For example, the agent might be asked to move certain objects between two locations. A common way to represent such scenes in e.g. the Computer Vision community is to make use of segmented and labeled images as depicted in Figure 1.1 b). While this representation is suitable for bench marking different models offline, it requires many hours of human labor to collect training data. Additionally, a model will have to be retrained whenever a new object is introduced. Another approach to this problem is a distributed, sensorimotor representation which can be acquired through interaction of the agent with each environment. As shown in Figure 1.1 c), a particular environment can then be represented in terms of its sensory attributes and action affordances. Whenever a new object is encountered, the agent can embed it in the same representation without the need to retrain the whole model.

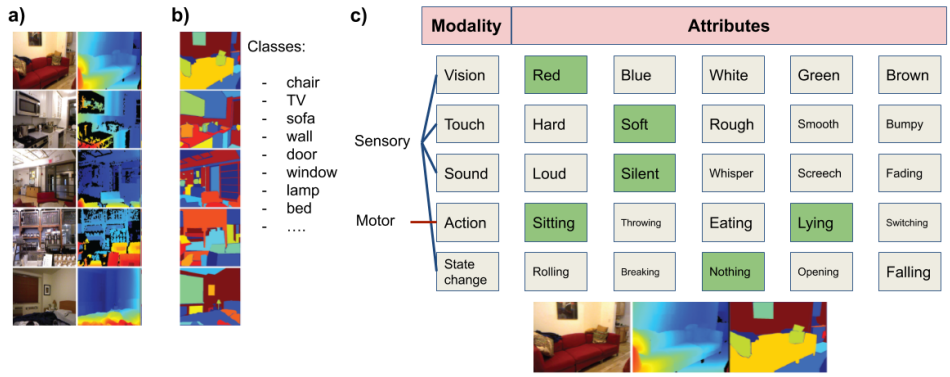


Figure 1.1: **a)** Images from the NYU-Depth data set [4], which consists indoor scenes recorded with RGB-D cameras. **b)** One way to represent this data is to segment and label objects and entities. **c)** A distributed representation of the same data that uses sensory and motor attributes to categories objects and entities. The scene in the bottom is represented by the attributes that are highlighted in green.

1.1 Sensorimotor contingencies

In the literature, this coupling between sensory and motor channels is called sensorimotor contingencies (SMCs) [3]. SMCs are statistical properties of action-perception loops that allow categorization of events, actions and objects and fluent interaction between an agent and its environment. In general, SMCs describe contingencies of different levels of complexity.

Body-internal SMCs First of all, motor output, such as force or muscle activation, is coupled to the internal sensory state of the body, such as joint angles or the position of limbs with respect to each other. In humans and other animals for example, the brain produces not only motor commands that are send to the limbs but also a so called efference copy which is a prediction of the sensation that a movement will induce. The efference copy is compared to the actual sensory outcome of the action [5]. A deviation from the prediction can lead to adjusting movements or to learning new behaviors.

Environmental SMCs Building on these foundations, action-induced effects in the environment can be coupled to specific movements and actions. In this case, the motor commands are related to e.g. changes in the visual field or the position of objects with respect to the actor. Assume a robot is given the task to move objects on a table. One way to go about this would be to hand-craft the identity of the objects and program the specific ways in which to move each object from A to B. However, this approach is not robust to failure and requires human involvement as soon as an unknown object appears in the scene. Instead, the robot can learn the SMCs that relate objects with certain properties,

such as round or tall, to actions applied to them, such as pushing and pulling. Once the SMCs are learned, the robot can choose the appropriate action for each object in order to move it from A to B, as e.g. shown in [6].

Social SMCs Finally, once body-internal SMCs and environmental SMCs are mastered, a cognitive system can go one step further and relate not only its own actions to its sensory input but also the actions of others. These social SMCs allow the agent to reason about the actions of others in its environment and to incorporate predictions about others' future behavior into its own action planning [7]. In humans and other primates, this coupling is measurable in the brain e.g. by to the existence of so called mirror neurons [8]. Mirror neurons are cells found e.g. in the motor cortex, where neural activity usually is correlated with the motor control of limbs. Mirror neurons however are not only active during movement planning and execution but also when the subject observes others performing the same movement. Thus, the cell "mirrors" or simulates others' movements as if the observer him -or herself was moving.

1.2 Requirements for embodied intelligence

In this work, we follow these ideas of embodied intelligence. An approach that relies on sensorimotor contingencies is by far not the only solution to the problem statements that we will introduce below. However, we belief that taking nature as an inspiration to build agents with some level of artificial intelligence is a first step to understanding the complexity of those problems.

In order to create an embodied system that can interact intelligently with both its environment and other agents, we require computational models that can represent all three types of SMCs: body-internal, environmental and social. Ideally, a single model class should be used to implement all three types as they need to interact with each other efficiently. Among others, the model class needs to be able to capture the following properties to represent all types of SMCs:

1. probabilistic - to cope with e.g. noise in sensory and motor channels
2. multimodal - to represent data from different modalities or actors
3. temporal - to represent state-action-effect relations over time
4. model latent variables - to cope with unknowns, such as others' intentions
5. unsupervised - to learn without human supervision

One type of machine learning models that is able to represent all of these requirements are generative models. As detailed in Chapter 2, generative models represent the joint probability distribution over a set of both continuous and discrete random variables. Once learned, they can be used to draw samples from the data generating distribution. Generative models are probabilistic by nature (property 1.), as they model probability distributions and can easily incorporate the dependencies between multiple modalities (property 2.) as they model joint distributions. In order to represent time (property 3.), a generative model

has to model the joint probability distribution over observations and actions at different time steps. As sensory observations are usually governed by unobserved, latent variables (property 4.), the generative model can be extended to model a joint distribution over both observed and latent variables. Often, the posterior over the latent variables is inferred with help of approximate inference techniques such as Monte Carlo methods and variational inference. Finally, as generative models represent a joint distribution over all variables, the data does not necessarily have to be labeled (property 5.). For example, generative models can learn a distribution over images without requiring class labels of the content of the images. In summary, generative models fulfill our requirements needed to represent all three types of SMCs as stated earlier. In the next Section, the generative approach to embodied intelligence will be related to the remaining content of this thesis.

1.3 A generative approach to action, prediction and interaction

In this work, we take a bottom-up approach to artificial intelligence. The main goal is to develop predictive models that allow a robot to interact both with the environment and with humans. In detail, without any assumptions about the robot system, the models should facilitate motion planning, make it possible to generate goal-directed actions in a changing environment and to represent, classify and anticipate human activity in a shared work space.

One requirement that these different problems have in common is to imagine future states given past observations. Since the world is usually not deterministic, at any given time point there exist a multitude of possible futures that need to be accounted for in order to make goal-directed decisions possible. Therefore we require probabilistic, temporal models of state-action-effect couplings, both for robot and human actions. The idea of embodied intelligence is highly related to the question of an optimal representation of the environment. Since we are aiming at building artificial systems, the representation needs to be of mathematical nature such that algorithms can be used to reason about the current and future states and to plan future actions.

Our mathematical tool of choice are therefore generative models. We demonstrate how generative models can be used for robot action generation and human activity understanding. Once these systems are in place, they can be applied in human-robot interaction scenarios. Here, the robot and human actions and their effects can be embedded in a common representation to allow for fluent and intuitive interaction.

The outline of the thesis is as follows: We first motivate and discuss the choice of generative models and the problem statements in the remainder of Part I. In Part II, the accompanying papers can be found, which detail the methodology and experiments. The remainder of this part begins with an introduction to generative models (Chapter 2). This is followed by a discussion of the challenges of robot learning (Chapter 3) and human activity modeling based on video data (Chapter 4). As human-robot interaction comes with its own challenges, we describe the setting in Chapter 5. Following a short conclusion in Chapter 6, a short summary of the accompanying papers is given in Chapter 7.

This introduction is intentionally kept on a high-level to develop the general idea that

relates the accompanying papers. Rich scientific work lies behind all of the introduced concepts which we can not make justice to in a short introduction. For topic relevant related work, we refer the reader to [9] and to the respective paper in Part II.

Chapter 2

Generative models

In this section, we will explain the concept of generative models in detail and introduce the model types that are used and further developed in the accompanying papers. In terms of notation, we assume $\mathbf{x}$ and $\mathbf{y}$ to be random variables. These variables could be of any type, such as real valued or categorical. An observed data point can be seen a sample from the data generating distribution $(\mathbf{x}_i, \mathbf{y}_i) \sim p^D_\theta(\mathbf{x}, \mathbf{y})$ and $\mathbf{D} = \{(\mathbf{x_i}, \mathbf{y_i})\}_{i=1:N}$ to be a set of N data points. A latent variable can be time dependent. In this case, we denote time step t by $\mathbf{x}_t$ and a time interval of length h by $\mathbf{x}_{t:t+h}$. If we want to describe all time steps t' before or equal to t, we write $\mathbf{x}_{t' \leq t}$. Equivalent notation is used for all time steps up to but not including t or after t. Additionally, we assume that there can exist unobserved latent variables $\mathbf{z}$ that are sampled from a prior distribution $\mathbf{z} \sim p_\theta(\mathbf{z})$. We denote the dependence of a probability distribution p on parameters θ by p_θ. We use p_θ to describe arbitrary probability distributions, i.e. the form of θ depends on a particular model, while we here discuss more general terms.

With the notation in place, we begin the discussion by clarifying the distinction between discriminative and generative models.

2.1 Discriminative and generative models

A goal of machine learning is to develop statistical algorithms that allow inference over the state of future data points given past observations. For many applications, such as classification or regression it is sufficient to describe the form of the target variable $\mathbf{y}$ as a function of the predictor variable $\mathbf{x}$. Given a dataset $\mathbf{D}$, the aim is therefore to determine the parameters θ, such that $p_\theta(\mathbf{y}|\mathbf{x})$ infers the value of $\mathbf{y}$ correctly for a given test data point $\mathbf{x} = \mathbf{x}^*$. This conditional probability $p_\theta(\mathbf{y}|\mathbf{x})$ is a **discriminative model**. In contrast, a **generative model** represents the joint probability distribution $p_\theta(\mathbf{x}, \mathbf{y})$ over all the variables. The advantage of this is that we can use generative models to draw samples that are similar to those drawn from the true data generating distribution. This ability to sample is useful for data augmentation or to *imagine* different future observations in a decision making scenario.

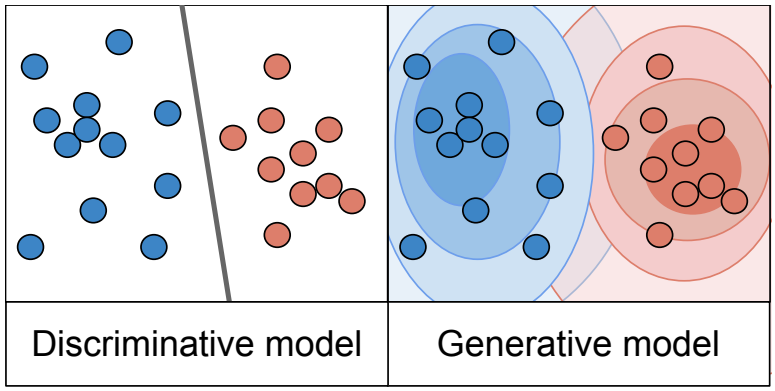


Figure 2.1: The circles represent observed data points that belong to one of two classes (red and blue). A discriminative model (left) seeks to find the optimal decision boundary (grey line) that enables classification of testing points. A generative model such as a Gaussian Mixture Model on the other hand represents the density over the $\mathbf{x}$ variable and encodes the class membership of a point in terms of distance to a mode of this density.

We visualize the difference in Figure 2.1. The circles represent observed data points that belong to one of two classes (red and blue). A discriminative model (left) seeks to find the optimal decision boundary (grey line) that enables classification of testing points. A generative model such as a Gaussian Mixture Model on the other hand represents the density over the $\mathbf{x}$ variable and encodes the class membership of a point in terms of distance to a mode of this density.

As an example for the application of the two model classes, assume that the random variable $\mathbf{x}$ describes an image and $\mathbf{y}$ the label of the image's content. A discriminative model $p_\theta(\mathbf{y}|\mathbf{x})$ can only be used to infer the label of a novel image. A generative model on the other hand can e.g. be used to sample images that contain content $\mathbf{y}$, making use of $p_\theta(\mathbf{x}|\mathbf{y})$.

A modern interpretation of generative models includes any model from which we can draw samples, such as Generative Adverserial Networks. While this an interesting area to explore, we focus on generative models in the traditional meaning of the term.

In this work, we mainly focus on two types of models, Gaussian Processes and Variational Autoencoders. While Gaussian Processes are usually classified as a discriminative model, we use them in a temporal manner, i.e. that we model the next state given the previous state. Unraveled over time, this forms a generative model over the state variable at time t given the state at time $t-1$. In the following, we will explain this in more detail and subsequently introduce the concepts of Deep Generative Models and Variational Autoencoders in particular.

2.2 Gaussian Processes as generative temporal models

Gaussian Processes are known to have a number of favorable properties, especially for robotics research. They are data efficient and model uncertainty over each test point by definition.

Gaussian Processes: A Gaussian Process (GP) defines a distribution over functions $\mathcal{GP}(f) = p_\theta(f)$ with $f : \mathbf{x} \to \mathbf{y}$. The distribution $p_\theta(f)$ is a Gaussian Process if for any finite set $\{\mathbf{x}_i\}_{i=1:N}$, where $\mathbf{f} = \{f(\mathbf{x}_i)\}_{i=1:N}$, the marginal distribution over that finite set $p_\theta(\mathbf{f})$ has a multivariate Gaussian distribution. This Gaussian distribution is parameterized by a mean function $\mu(\mathbf{x})$ and a covariance (or kernel) function $K(\mathbf{x}, \mathbf{x}')$. Usually, the mean function is assumed to be zero.

Gaussian Processes commonly assume the following data generative process:

$$\mathbf{y} = f(\mathbf{x}) + \varepsilon, f \sim \mathcal{GP}(f|\mathbf{0}, K), \varepsilon \sim \mathcal{N}(0, \sigma) \tag{2.1}$$

To make predictions for a test point $\mathbf{x}^*$ given observed data D, we can integrate over the functions

$$p_\theta(\mathbf{y}^*|\mathbf{x}^*, D) = \int_f p_\theta(\mathbf{y}^*|\mathbf{x}^*, f, D) p_\theta(f|D) \tag{2.2}$$

For an extensive introduction to Gaussian Processes, e.g. on how to tune the kernel hyper-parameters, we refer the reader to [10].

It becomes apparent, that the vanilla definition of GPs is not generative because we do not model a distribution over $\mathbf{x}$. However, when we apply GPs to temporal data, we can phrase the problem of prediction as an autoregressive generative model [11]. The generative process becomes now

$$\mathbf{x}_t = f(\mathbf{x}_{t-1}) + \varepsilon, f \sim \mathcal{GP}(f|\mathbf{0}, K), \varepsilon \sim \mathcal{N}(0, \sigma), \tag{2.3}$$

and the predictive distribution

$$p_\theta(\mathbf{x}_t^*|\mathbf{x}_{t'<t}, D) = \int_{f, \mathbf{x}_{t-1}} p_\theta(\mathbf{x}_t^*|\mathbf{x}_{t-1}, f, D) p_\theta(f|D) p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_{t'<t-1}, D). \tag{2.4}$$

In this way, the autoregressive GP can be interpreted as a generative model. We use GPs for robot learning for motor control as introduced in Chapter 3 and in a reinforcement learning setting for physical human-robot interaction as presented in Chapter 5.

With the advances of deep neural networks, another class of generative models have emerged, namely Deep Generative Models. We will introduce these models in the next section and put a special emphasis on Variational Autoencoders.

2.3 Deep Generative Models

As generative models represent joint distributions, often over both observed and latent variables, the inference over model parameters and latent variables can be problematic or

even intractable. Especially in high-dimensional input spaces such as images it has proven difficult to design a feature space that is rich enough to explain the data at hand. Deep generative models have been successfully used to overcome these problems. On the one hand, they often treat inference as a black-box, on the other hand, deep neural networks are known for their representation learning capabilities [12].

In general, deep generative models are trained with help of back-propagation techniques to learn a probability distribution that is as close as possible to the data generating distribution. A common approach is to sample a noise variable from a simple distribution, such as a standard normal distribution, and to transform this sample with help of neural network architectures to resemble a sample from the data generating distribution.

Traditionnaly, generative models represent probability distributions with help of parameters, that are fixed after training. In the case of a Gaussian Mixture Model e.g., these parameters would be the mixture weights as well as means and variances of the Gaussians. For example, a single Gaussian fitted to data set D would have the following form:

$$\text{Generative model: } \mathbf{x} \sim p_{\mu,\sigma}(\mathbf{x}) = \mathcal{N}(\mu,\sigma), \quad (\mu,\sigma) \sim p_\theta(\mu,\sigma|D). \tag{2.5}$$

Deep generative models on the other hand, assume only the form of the output distribution, e.g. independently and identically distributed (iid) Gaussian distributions, whose parameters are determined by the transformations of the noise variable. Thus, a deep generative model trained on the same data D as the model in Equation 2.5 would have the form:

$$\text{Deep generative model: } \mathbf{x} \sim p_{(\theta_\mu,\theta_\sigma)}(\mathbf{x}) = \mathcal{N}(\mu(\mathbf{z},\theta_\mu),\sigma(\mathbf{z},\theta_\sigma)), \quad \mathbf{z} \sim \mathcal{N}(0,1) \tag{2.6}$$

In contrast to the traditional inference procedure in Equation 2.5, the parameters of the Gaussian in Equation 2.6 are described by the flexible neural network functions $\mu(\cdot,\theta_\mu)$ and $\sigma(\cdot,\theta_\sigma)$.

There exist a number of deep generative models which differ in modeling assumptions and inference techniques. We can distinguish between four types of deep generative model: 1) Variational Autoencoders [17, 28], 2) Generative Adverserial Networks [15], 3) Autoregressive generative models [16] and 4) Flow-based generative models [17, 18]. Since we will only make use of variations of Variational Autoencoders, we will explain the ideas behind this model class in detail in the next section.

Variational Autoencoders

Variational Autoencoders (VAE) are deep latent variable models that employ neural networks to infer an approximate posterior over latent variables and to generate data samples. We will begin this section with describing the assumed generative process and derive a variational inference formulation to determine the approximate posterior. Given these foundations, we explain how Variational Autoencoders perform inference often more efficiently.

Assume the following data generating process

$$\mathbf{x} \sim p_\theta(\mathbf{x}|\mathbf{z}), \quad \mathbf{z} \sim p_\theta(\mathbf{z}), \tag{2.7}$$

where $\mathbf{x}$ is the observed variable which depends on a latent variable $\mathbf{z}$. Often we assume that $\mathbf{z}$ is of a lower dimension than $\mathbf{x}$, which makes it desirable to infer the posterior distribution $p_\theta(\mathbf{z}|\mathbf{x})$. For example, in a Gaussian Mixture Model $\mathbf{z}$ can represent the mixture assignment of a data point $\mathbf{x}$ which can be used for classification. In other applications, $\mathbf{z}$ might represent the mapping of $\mathbf{x}$ onto a lower-dimensional space that encodes the data generating factors such as color, position and lightning in an image. However, often it is intractable to infer the structure of the posterior $p_\theta(\mathbf{z}|\mathbf{x})$. In this case, approximate inference methods can be applied to approximate the posterior either exactly by sampling (Monte Carlo methods) or approximately by optimization (variational inference). We will here focus mostly on the latter technique and exemplify it with help of mean field approximation.

Let us assume that each data point in $\mathbf{X} = \{\mathbf{x}_i\}_{i=1:N}$ was generated from a corresponding latent variable $\mathbf{Z} = \{\mathbf{z}_i\}_{i=1:N}$. In order to determine an approximate posterior distribution $q_\phi(\mathbf{Z})$, we assume a factorized distribution $q_\phi(\mathbf{Z}) = \prod_i q_\phi(\mathbf{z}_i, \lambda_i)$, where each factor i depends on local variational parameters λ_i, with $\lambda = \{\lambda_i\}_{i=1:N}$. To minimize the distance between the true posterior $p_\theta(\mathbf{Z}|\mathbf{X})$ and $q_\phi(\mathbf{Z}, \lambda)$, variational inference (VI) makes use of the log likelihood of the data, which can be shown to have the following form [12]

$$\log p_\theta(\mathbf{x}) = \mathbb{E}_{q_\phi(\mathbf{z};\lambda)}\left[\log \frac{p_\theta(\mathbf{x},\mathbf{z})}{q_\phi(\mathbf{z};\lambda)}\right] + D_{KL}(q_\phi(\mathbf{z},\lambda)||p_\theta(\mathbf{z}|\mathbf{x})). \tag{2.8}$$

, where D_{KL} is the Kullback-Leibler (KL) divergence between two distributions

$$D_{KL}(q_\phi(\mathbf{z})||p\theta(\mathbf{z})) = -\int q_\phi(\mathbf{z}) \log \frac{p_\theta(\mathbf{z})}{q_\phi(\mathbf{z})} d\mathbf{z}. \tag{2.9}$$

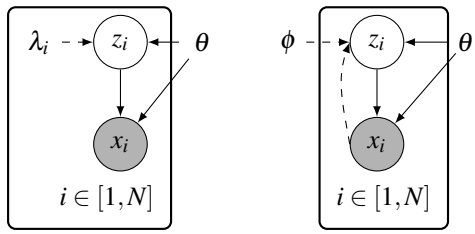
To decrease the distance between $p_\theta(\mathbf{z}|\mathbf{x})$ and $q_\phi(\mathbf{z}, \lambda)$ we need to minimize the KL divergence between the two, which is the second term in Equation 2.8. Since the KL divergence is a distance measure it is always positive and only zero when $p = q$. Thus, minimizing the KL divergence is equivalent to maximizing the first expectation in Equation 2.8, which is commonly known as the *Evidence Lower BOund (ELBO)* $\mathcal{L}(\lambda)$:

$$\begin{aligned} \log \int p_\theta(\mathbf{x},\mathbf{z}) d\mathbf{z} &= \log \int \frac{p_\theta(\mathbf{x},\mathbf{z}) q_\phi(\mathbf{z};\lambda)}{q_\phi(\mathbf{z};\lambda)} d\mathbf{z} = \log \mathbb{E}_{q_\phi(\mathbf{z};\lambda)}\left[\frac{p_\theta(\mathbf{x},\mathbf{z})}{q_\phi(\mathbf{z};\lambda)}\right] \\ &\geq \mathbb{E}_{q_\phi(\mathbf{z};\lambda)}\left[\log \frac{p_\theta(\mathbf{x},\mathbf{z})}{q_\phi(\mathbf{z};\lambda)}\right] \equiv \mathcal{L}(\lambda). \end{aligned} \tag{2.10}$$

While VI is a powerful tool, it suffers from the need to determine local parameters $\lambda = \{\lambda_i\}_{i=1:N}$ for each data point, as shown in Figure 2.2 (a). This becomes impractical for a large number of training points and requires expensive inference even at test time.

Variational Autoencoders (VAEs) circumvent this problem with help of a parameterized function that maps each data point to its corresponding approximate posterior distribution. This parameterized function is often a deep neural network, the inference network, with parameters ϕ, $q_\phi(\mathbf{z}|\mathbf{x})$. For example, if we assume $q_\phi(\mathbf{z}|\mathbf{x})$ to consist of iid Gaussian distributed variables, then

$$\mathbf{z}_i \sim q_\phi(\mathbf{z}_i|\mathbf{x}_i) = \mathcal{N}(\mu(\mathbf{x}_i, \phi_\mu), \sigma(\mathbf{x}_i, \phi_\sigma)) \tag{2.11}$$



(a) VI (b) VAE

Figure 2.2: The inference procedure in a simple latent variable model using variational inference (a) and variational autoencoders (b). Variational approximations are indicated by dashed lines.

where $\phi = (\phi_\mu, \phi_\sigma)$. Likewise, a generative network learns a paramterized mapping from the latent space to the data space $p_\theta(\mathbf{x}|\mathbf{z})$. Note that both ϕ and θ are the parameters of the neural networks and not the parameters of the probability distributions q and p as discussed in relation to Equation 2.6. As shown in Figure 2.2 (b), VAEs do not require the local variational parameters anymore but rely on the learned mapping between the two spaces. To train the inference and generative network, the model is trained with back-propagation to optimize the ELBO

$$\mathscr{L}(\phi,\theta) = \frac{1}{N}\sum_{i=1:N} \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_i)} p_\theta(\mathbf{x}_i|\mathbf{z}) - D_{KL}(q_\phi(\mathbf{z}|\mathbf{x}_i)||p_\theta(\mathbf{z})), \tag{2.12}$$

where $p_\theta(\mathbf{z})$ is the prior over $\mathbf{z}$. Back-propagation with low variance is possible due to the so called reparamerization trick. For example, when $\mathbf{z}$ is Gaussian distributed, it is sampled according to $\mathbf{z} = \mu + \sigma * \varepsilon, \varepsilon \sim \mathcal{N}(0,1)$. The expectation in Equation 2.12 is approximated with Monte Carlo samples from q_ϕ.

The term variational autoencoder originates partly from the formulation as variational inference in Equation 2.12 and partly from their similarity to autoencoders. Since the dimensionality of $\mathbf{z}$ is usually chosen to be smaller than the dimensionality of $\mathbf{x}$, the mapping $\hat{\mathbf{x}} = p_\theta(q_\phi(\mathbf{x}))$ resembles the structure of an autoencoder with a stochastic bottleneck layer.

For a more detailed review on the advances in amortized inference, e.g. VAEs, we refer the reader to [12]. In this work, we use VAEs to model human behavior in a predictive manner both in terms of continuous time series features and semantic labels as introduced in Chapter 4.

Chapter 3

Self-learning for motor control

Motor control for robotics is a broad topic which we will not be able to make justice to in this chapter. We will instead focus on the motivation for using Bayesian and generative models for self-learning of motor control. In detail, we describe how we collect data for self-learning in Section 3.1, which is followed by an introduction to predictive learning and learning under uncertainty in Section 3.2 and Section 3.3 respectively.

3.1 Learning by exploration

Our approach to develop algorithms for robot motor control learning is to assume as little as possible about the system. Instead of assuming e.g. kinematic chains we aim at learning SMCs between the current state $\mathbf{s}_t$ of the system, the current action $\mathbf{a}_t$ and the next state $\mathbf{s}_{t+1}$ that is a consequence of this action, see e.g. [19]. Just as a new born infant, we would like the robot to explore its own capabilities and learn the SMCs associated with its own actions. To train generative models that are capable of this, we require triplets of the form $(\mathbf{s}_t, \mathbf{a}_t, \mathbf{s}_{t+1})$. Mimicking the motor babbling behavior of newborns, we make use of random exploration under safety constraints. This means, that we instruct the robot to apply random actions in a predefined state and action space and record the data points $(\mathbf{s}_t, \mathbf{a}_t, \mathbf{s}_{t+1})$. In our work, we define the state $\mathbf{s}_t$ to encompass e.g. the angle and angular velocity of relevant joints as well as torque measurements. When interacting with an object of interest, the state does also contain relevant information about e.g. the position and velocity of the object. The action $\mathbf{a}_t$ can be either defined as torque or angular velocity commands. In contrast to the state, which is subject to noise and environmental influences, the actions can be treated as a deterministic variables instead of random variables as they are determined by the robot's motor commands.

Once a generative model of the SMCs is learned, there exist two methods for goal-directed action generation. On the one hand, the model can be used to plan a trajectory towards the goal. On the other hand, we can apply model-based reinforcement learning to learn a policy that allows automatic action selection. We explore both of these ideas in the Paper A and B respectively.

3.2 Predictive learning

Interaction with the environment and with humans requires a robot to react fast to changes. Therefore, we favor predictive control over reactive control. Imagine a robot is supposed to shake hands with a human. While the robot could wait in a reactive manner until the human has lifted their hands to the initial position before initiating its own movement, this behavior would be rather frustrating for the human partner. Instead, we aim at robot systems that choose actions according to a predictive model of the future state. This enables the robot to initiate the movement soon after the human started to move.

We can implement predictive control with help of a forward model

$$\mathbf{s}_{t+1} = \mathbf{s}_t + \Delta\mathbf{s}_t, \quad \Delta\mathbf{s}_t \sim p_\theta(\Delta\mathbf{s}_t|\mathbf{s}_t, \mathbf{a}_t)$$

that predicts a distribution over the future state given the measured past state and a deterministic action. Instead of directly predicting the next state, the forward model predicts the change of the state $\Delta\mathbf{s}_t = \mathbf{s}_{t+1} - \mathbf{s}_t$ caused by the action. As discussed in Section 2.3, we view this model as a generative model with distributions over subsequent states while we treat actions to be deterministic. The parameters of the forward model are trained on the state-action pairs $(\mathbf{s}_t, \mathbf{a}_t, \mathbf{s}_{t+1})$ that were collected with help of random exploration. In the case of GPs, as mainly used for self-learning in this work, the hyperparameters of the kernel function are optimized with maximum likelihood estimation.

Next to decreased reaction time, predictive acting has several other advantages. For example, if the outcome of an action does not match the predicted state, the model might be inaccurate and might be in the need of an update. However, in case the system faces noisy sensory signals, this deviation might also arise due to the uncertainty in the environment. To differentiate between these two cases we require an explicit representation of uncertainty as we discuss next.

3.3 Incorporating uncertainty

Uncertainty in a robot's environment can arise due to many factors. A primary source of uncertainty in the sensorimotor system originates from noisy sensors. Therefore we require generative models that are able to represent this uncertainty for example in form of variances. Importantly, different sensory dimensions might have different degrees of variance. For example, the angle measurements of the shoulder joint will be less noisy than the measurements of the wrist joint as the shoulder is less flexible and does not depend on the position of other rigid joints. Gaussian Processes are a natural way to represent this uncertainty. On the one hand, the kernel function expresses uncertainty about regions in the data space that are far from any training data point. On the other hand, the σ noise term in Equation 2.1 can be adjusted to account for the noise level in each dimension. In Paper A we present how these mechanisms can be used for adaptive control.

When applying VAEs instead of GPs, care can be taken to tune the output variance of the generative network. However, this variance estimate has been shown to not reliably

represent represent uncertainty about unknown areas of the data space [20]. Uncertainty estimation methods such as variational dropout [21] could be used to mitigate this problem.

As soon as humans are part of the robot's environment, a new source of uncertainty is introduced. Since the human is acting independently of the robot, there exists uncertainty about the human's future location and actions. We will introduce our approach to model these activities in the next section.

Chapter 4

Challenges and tasks of human activity modeling

In this work we choose a top-down approach to social SMCs, which means that we first develop generative, predictive models of human activity which are then mapped to and integrated with the robot's own SMCs. Human behavior is intrinsically complex. To understand and predict human actions accurately is not only challenging for machines, but also for humans. In order to predict actions and movements, a computational model needs to represent the task, the environment, the human's preferences and more. In this chapter we will discuss some of these challenges that are relevant for our task setting of human-robot interaction in Section 4.1. The computer vision community is working to solve a number of different tasks when it comes to human activity modeling, which we will detail in 4.2. However, the research in computer vision is often focused on static, clean data sets, while we here advocate systems which can be applied under real-world requirements as discussed in Section 4.3.

4.1 Challenges of human activity modeling

Human activity modeling faces many challenges, some of which are shared with other areas of robotics such as the problem of partial observations, and some of which are unique to modeling other agents, such as intentions and beliefs. We will here list those problems that we address in the accompanying papers.

Latent factors

Latent random variables in a machine learning model usually represent unobserved entities such as data generating factors, factors that we can not directly measure or random noise. In case of human activity, random factors include those factors that drive the human's behavior such as intentions, beliefs, preferences and past experiences. Imagine a kitchen scenario where a human subject takes the green instead of the red knife to cut an apple.

This choice could be driven by preference for the color green, the knowledge that the green knife has been sharper the last time that the task was performed or the superstitious belief that cutting a red apple with a red knife brings bad luck. If a robot is supposed to learn how to cook an apple pie from observing the human, the choice of knife does not matter in the case of the belief or preference, but would matter in the case of acquired knowledge about the sharpness. Thus, oftentimes we can treat latent factors as latent variables that can be used to explain variations in behavior but their meaning must not directly inferred. It is challenging to determine which latent factors are task-relevant and which can be ignored for general human activity modeling and robot learning.

Many possible futures

In a non-deterministic world, a single past has many possible futures. Due to the latent factors discussed above and environmental influences, human activity is basically never deterministic and therefore we can assume that given an observation of past human activity there exists an infinite amount of possible futures. While the number of possible future high-level actions, such as whether to take the red or the green knife, is constrained by the environment and the human's capabilities, the amount of possible motion trajectories in a three-dimensional space of real numbers is infinite. Obviously, some trajectories will be more likely than others, especially in goal-directed movements. Therefore we require probabilistic models than represent distributions over future actions and movements and which can be used to assess the likelihood of future observations.

Sensory noise

In order to interact with a human in a shared work space, the robot requires an understanding of the position of the human in 3D space. Thus, it is not enough to merely model human activity on the basis of image or video data. Instead, we also need to make use of depth sensors or motion capture technology to estimate the 3D position and pose of the human with respect to the environment. The quality of most low-end sensors, that would be available even in a users kitchen and not only in a research lab, is still rather poor. The estimated joints can arbitrarily change position or are flickering. Any joint that is not in the view field of the camera can lead to unexpected, violent jiggling of the joint in the recording. This sensory noise poses not only a problem for e.g. action classification but also for learning predictive models of human motion trajectories, which might either follow the unnaturally looking data distribution or regress to an uninformative mean prediction.

Partial knowledge

When a model, that has been trained on labeled human activity data, is tested in an online human-robot interaction application, the human might perform actions that have not been part of the training data. This partial knowledge about the capabilities of the human and the uncertainty about unknown possible futures requires adaptable models that can be used to 1) detect when a novel observation is made and 2) incorporate this observation into

the already acquired model with e.g. continual learning. While we only treat the case of missing labels in our work, we see a need for further development on continual learning and novelty detection.

Spatio-temporal complexity

To model interaction with the environment, requires the model to represent many variables that interact both in the spatial and temporal dimension. Take the apple pie baking as an example. Here the variables would be all the kitchen utensils and ingredients needed for baking and the human itself. The state of each of the object variables, such as affordances and position on the table, as well as their relation to the human need to be accounted for. Additionally, the joint positions of the human and the action that is currently performed have to be incorporated. The correlation between all of these variables and their state has to be modeled not only for the current observation but in relation to the temporal progression of the task.

4.2 Tasks in human activity modeling

Human activity modeling is an import area of research in the computer vision community. The different approaches differ not only in task formulation, as presented below, but also in the data types used to represent human activity. The data types include among others still images, video recordings, depth image and video data, inertia sensors and motion capture recordings. In the following, we will represent an observation of any kind at time step t with $\mathbf{x}_t$. These observations can be accompanied with labels of actions, action hierarchies, affordances and more, which we will here denote with $\mathbf{y}_t$. In this work, we focus on skeletal recordings of human motion in the 3D euclidean space, which in some cases also incorporate information about the 3D position and state of objects. When required by the model, we use available label data as well.

In the following we introduce six tasks that are concerned with certain parts of human activity modeling. We will first discuss motion prediction and generation which are both modeling mainly the development of $\mathbf{x}_t$ over time. Subsequently, we will introduce action classification, prediction, detection and anticipation, which focus on inferring $\mathbf{y}_t$ from $\mathbf{x}_{t' \leq t}$ at different time points.

Motion prediction

Motion prediction is concerned with inferring the most likely future trajectory in a time window with duration h from the next state $\mathbf{x}_{t+1}$ to a future state $\mathbf{x}_{t+h+1}$. This prediction is usually based on the past observations $\mathbf{x}_{t' \leq t}$. Thus, the task of motion prediction is concerned with determining the parameters θ that minimize the predictive error of the expectation $f(\mathbf{x}_{t' \leq t}, \theta) = \mathbb{E}_{p_\theta(\mathbf{x}_{t+1:t+h+1} | \mathbf{x}_{t' \leq t})}[\mathbf{x}_{t+1,t+h+1}]$.

Motion generation

In contrast to motion prediction, motion generation aims at generating different possible future trajectories $\mathbf{x}_{t+1,t+h+1}$. Therefore, instead of just determining an expected value, we are interested in modeling the whole distribution $p_\theta(\mathbf{x}_{t+1:t+h+1}|\mathbf{x}_{t'\leq t})$.

Action classification

Action classification assumes that an observed sequence of length T has only a single underlying label, i.e. only a single action was performed. The goal of this task is determine the correct label, by minimizing the training error of $p_\theta(\mathbf{y}_T|\mathbf{x}_{1:T})$ with respect to parameters θ.

Action prediction

Instead of waiting until the whole sequence has been observed as is the case of action classification, action prediction is concerned with determining the label as soon as possible. Thus, it is concerned with minimizing the classification error of $p_\theta(\mathbf{y}_t|\mathbf{x}_{1:t})$ at each time step.

Action detection

When an observed sequence contains not only one but several different actions, it is important to detect the change of action. Action detection is accomplished by minimizing the training error of $p_\theta(\mathbf{y}_t|\mathbf{x}_{1:t},\hat{\mathbf{y}}_{1:t-1})$, where $\hat{\mathbf{y}}_{1:t-1}$ are the inferred labels up to time $t-1$. A different approach could be to directly attempt to determine a change of action by minimizing the training error of $p_\theta(\mathbf{y}_t! = \hat{\mathbf{y}}_{t-1}|\mathbf{x}_{1:t},\hat{\mathbf{y}}_{t-1})$.

Action anticipation

Finally, action anticipation is supposed to determine future actions given the observed past. Thus, the task is here to determine the model $p_\theta(\mathbf{y}_{t+1}|\mathbf{x}_{1:t},\mathbf{y}_{1:t})$ or $p_\theta(\mathbf{y}_{t+1}|\mathbf{x}_{1:t},\hat{\mathbf{y}}_{1:t})$ that best describes the training data.

4.3 Real-world employment

A robot system, that is supposed to interact with a human in real time, requires all of the tasks described in Section 4.2 to be solved simultaneously. Likewise, the model needs to be able to cope with the challenges discussed in Section 4.1. While most work in computer vision focuses on a single task and is evaluated in an offline setting, we advocate a model class that can solve multiple tasks simultaneously and online while being flexible enough to address the challenges of human activity modeling. As presented in Paper E, we have developed a generative model framework based on Variational Autoencoders that does address these real-world requirements. As a precursor, we present two works in Papers C

and D that are concerned with Motion prediction and Motion generation respectively. The motion prediction model relies on a deterministic encoder-decoder structure, which is not generative, but which builds the basis for the temporal VAE model for motion generation presented in Paper D.

In the next section, we introduce how this generative model framework can be used to solve physical human-robot interaction tasks.

Chapter 5

Generative models for human-robot interaction

In the last chapters we introduced the concept of generative models and described how these are useful for self-learning of body-internal and environmental SMCs in robots and for human activity modeling. In this final chapter, we want to focus on how to combine these ideas to learn social SMCs which are required for human-robot interaction. We begin the discussion by describing how a human partner can be integrated into the robot learning process either passively in form of an extended state representation or actively through learning by demonstration (Section 5.1). This will be followed by a discussion of how to implement the actual interaction in terms of predictive social SMCs (Section 5.2).

5.1 From learning to act to learning to interact

Robot learning of SMCs in an environment that is shared with a human can be accomplished in two ways. On the one hand, the robot can encode the human passively in terms of its own SMCs. For example, in Paper B, the robot interacts with a human by controlling a ball that is rolling on a plank which is jointly held by the robot and the human. In this case, it is sufficient to incorporate the human in form of the force that he or she is exerting on the object.

On the other hand, the robot can learn from the human through observation or direct teaching [22]. Learning from observation requires the robot to already have established a number of body-internal and environmental SMCs. Observing the human performing a task implies to map the actions of the human to the robots embodiment. For example, assume that the robot has learned how to move, how to grasp a knife and cut objects and observes a human cutting an apple. In order to learn this task, the robot needs to decipher the humans behavior action by action and map the human's actions to its own action capabilities. To find such a mapping is called the correspondence problem. We will discuss this shortly together with using the learned mapping in a predictive manner in the next section.

5.2 Interaction through mapping and prediction

The correspondence problem [23] aims at determining a mapping between an observed human body and the robot's own embodiment. If the degree-of-freedom of the robot are not identical a human's degrees-of-freedom, this mapping can be learned with machine learning techniques, e.g [24]. How can we realize such a mapping, even if it is not one to one, with help of generative models? Given access to a data set of corresponding poses, where the human is mirroring the robot's pose or vice versa, one could model the joint distribution $p_\theta(\mathbf{s}_t^R, \mathbf{s}_t^H)$, where $\mathbf{s}_t^R$ and $\mathbf{s}_t^H$ are the poses of the robot and the human respectively. To acquire such a data set is cumbersome as we might require a large number of correspondences. A different approach would be to first gather data of the robot and the human independently and to model two generative models with latent variables $p_\theta^R(\mathbf{s}_t^R, \mathbf{z}_t^R)$ and $p_\theta^H(\mathbf{s}_t^H, \mathbf{z}_t^H)$. Finally, a smaller number of corresponding data points can be used to establish a smooth mapping between the two latent spaces $\mathbf{z}_t^R = f(\mathbf{z}_t^H)$. Once a correspondence map has been established, it can be used to encode the human's actions in terms of the robot's embodiment. For example, assume the robot knows how to execute an action but not how to determine the action underlying a human's movements. With help of the correspondence mapping, the human's trajectories can be identified by relating them to the robot's action repertoire.

One might also imagine a scenario where a mapping between embodiments is replaced by a mapping between action capabilities. In most high-level tasks, the exact execution of an action is of less importance than the correct execution of the task. Especially in collaborative tasks, the order of actions and interactions is of greater importance than exactly mimicking human trajectories. Sharing a task requires predictive models of what actions the human partner will perform next. Additionally, sharing a work space requires predictive models of the possible future human pose to avoid e.g. collisions. As introduced in Chapter 4, we developed predictive, generative models that can accomplish both levels of prediction.

Finally, based the environment as well as body-internal, environmental and social, predictive SMCs, the robot can plan actions that are in accordance with a given task. If the task is to bake an apple pie together, the robot might infer that the human will add the eggs next and predict the required trajectories of the arms and hands. This allows the robot to choose another sub-task, such as adding sugar, and to plan motion trajectories to the sugar pot that do not conflict with the future movements of the human. In this way, we imagine complex human-robot interaction tasks to be executed efficiently while keeping the safety of the human in mind.

Chapter 6

Conclusion and Future Work

In this thesis, we show how robot learning, human activity understanding and physical human-robot interaction can be implemented with help of generative models of sensorimotor contingencies. Interacting with humans in a shared environment is complicated by many factors. Making use of sensorimotor contingencies and generative models removes the need for explicit representations of actions and objects or hard-coded behaviors. Furthermore, a probabilistic machine learning approaches can express uncertainty in noisy environments while latent variables can encode unobserved factors such as intentions.

Future work includes many directions. All three areas, robot-learning of SMCs, modeling human activity and combining the two approaches to allow for efficient HRI need to be explored further. One point of interest is how to achieve adaptive systems, that can adjust to unexpected changes, identify unknowns and be adapted online to capture novel data points.

Chapter 7

Summary of papers

A Self-learning and adaptation in a sensorimotor framework

In this paper, we present an approach to robot self-learning of goal-directed motor control. The task setting that we envision here to reach a goal state represented which is represented by sensorimotor signals of the robot, e.g. a certain joint angle configuration. We present a general framework to autonomously achieve the task of finding a sequence of actions that result in the desired state. To acquire autonomous behavior, the robot is learning forward models that describe how to relate the the current and action to a future state. Initial training data of sensorimotor patterns is acquired without supervision through the robot's interaction with its environment.

Gaussian processes (GP) are used to learn the sensorimotor mapping. The GPs are viewed as generative forward models that represent a distribution over each future state dimension given the current state and the current action. We make use of automatic relevance determination to systematically select task-relevant sensory and motor components in the higher dimensional input space.

We propose an incremental GP learning strategy, which discerns between situations, when an update or an adaptation must be implemented. When the prediction error of the forward model is high, but the input space close to the current state is well-explored, the model needs to be adapted. However, when the current state is in a novel area of the input space, the model merely needs to be updated.

To enable long-term planning and generating a sequence of states that lead to a given goal, we exploit Rapidly exploring Random Tree (RRT*) algorithm. In this tree, a gradient-based search finds the optimum action to steer to a neighboring state in a single time step.

Our experimental results prove the suitability of the proposed framework to learn a joint space controller with high data dimensions (10×15). We demonstrate that the system can learn goal-directed motor control based on a short training phase (less than 12 seconds) and operate in real-time. The choice of GPs demonstrates rapid adaptation capabilities to novel sensory states. These are enabled by the automatic relevance determination and the uncertainty estimates of the model.

Published at the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2016.

B A sensorimotor reinforcement learning framework for physical human-robot interaction

Modeling of physical human-robot collaborations is generally a challenging problem due to the unpredictive nature of human behavior. To incorporate the human into the robot's decision making, we assume that it is possible to measure the humans actions in terms of the robot's sensory channels, e.g. force signals. The robot learns to collaborate with the human from its own sensorimotor experiences in an unsupervised manner. We begin by collecting sensorimotor data in interaction with the human. The data is recorded through motor babbling within the collaborative environment.

We present a data-efficient reinforcement learning framework which enables a robot to learn how to collaborate with a human partner. The uncertainty in the interaction is modeled using Gaussian processes (GP). GPs are trained to represent a forward model that models a distribution over the next state given the current state and action. With help of this model, we train a GP that implements an action-value function (Q-function). After training, the action-value function serves as a policy that directs the robot's actions while executing a collaborative task with a human. Optimal action selection given the uncertain GP model is ensured by Bayesian optimization.

We apply the framework to a scenario in which a human and a PR2 robot jointly control the ball position on a plank based on vision and force/torque data. The human's actions are taken into account through the force/torque data that is measured at the wrist of the robot. This data contains information about the magnitude and direction of the force that the human exerts on the plank.

Our experimental results show the suitability of the proposed method in terms of fast and data-efficient model learning and optimal action selection under uncertainty. We demonstrate that role sharing between the partners can be accomplished by rewarding low torque measurements, i.e. the human does not have to carry the whole workload.

Published at the IEEE International Conference on Robotics and Automation (ICRA), 2016.

C Deep representation learning for human motion prediction and classification

In this work, we consider the task of skeletal human motion prediction in 3D. Generative models of 3D human motion are often restricted to a small number of activities and can therefore not generalize well to novel movements or applications. Especially methods that are based on recurrent neural networks are often action-specific, i.e. one model is able to predict sequences of a *walking* human while another model is trained to predict sequences of *jumping*.

In this work we propose a deep learning framework for human motion capture data that learns a generic representation from a large corpus of motion capture data and generalizes well to new, unseen, motions. Using an encoding-decoding network that learns to predict future 3D poses from the most recent past, we extract a feature representation of human motion. Instead of single time steps, we train the model to encode a time window of poses and to decode the corresponding future time window.

Most work on deep learning for sequence prediction focuses on video and speech. Since skeletal data has a different structure, we present and evaluate different network architectures that make different assumptions about time dependencies and limb correlations. We compare (1) a network with plain feedforward connections, (2) a time convolutional neural network with filters of different time scales, and (3) a hierarchical approach that encodes the limb structure of the human body.

To quantify the learned features, we use the output of different layers for action classification and visualize the receptive fields of the network units. Our method outperforms the recent state of the art in skeletal motion prediction even though these use action specific training data. Our results show that deep feedforward networks, trained from a generic mocap database, can successfully be used for feature extraction from human motion data and that this representation can be used as a foundation for classification and prediction.

While this encoding-decoding model is not a generative model in the mathematical sense, it can be extended by a variational autoencoder formulation.

Published at the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

D Anticipating many futures: Online human motion prediction and generation for human-robot interaction

Fluent and safe interactions of humans and robots require both partners to anticipate the others' actions. The bottleneck of most methods is the lack of an accurate model of natural human motion. In real-time applications, this model must not only be efficient when making predictions but also be able to cope with sensory noise and to take many possible futures into account. In this work we consider human skeletal poses that are obtained from RGB depth images in real-time. The goal is to design a generative model that can be used to generate possible future motion trajectories.

To this end, we present a conditional variational autoencoder that is trained to predict a window of future human motion given a window of past frames. We formulate a temporal encoder-decoder structure with two latent variables. This enables us to sample future human motion trajectories that are in accordance with the past observation.

Using skeletal data obtained from RGB depth images, we show how this unsupervised approach can be used for online motion prediction for up to 1660 ms. Additionally, we demonstrate how to make use of the predictions to classify the target location of a grasping motion. Other approaches to online target prediction usually require target specific training data and model each target directed trajectory separately. Instead, our generative model allows us to predict the future trajectory and classify the target based on this prediction. Our approach is able to predict the target online within the first 300-500 ms after motion onset without the use of target specific training data.

Finally, we investigate how movements and kinematic cues are represented on the learned low dimensional manifold by considering the legibility and predictability of natural and unnatural reaching movements.

Published at the IEEE International Conference on Robotics and Automation (ICRA), 2018.

E A Probabilistic Semi-Supervised Approach to Multi-Task Human Activity Modeling

Human behavior is a continuous stochastic spatio-temporal process which is governed by semantic actions and affordances as well as latent factors. For computer vision systems that are designed to model human activity this poses several challenges. Next to sensory noise and only partially observable environments, a computer vision system should be able to cope with the fact that any observation up to the current state can have many possible futures.

The computer vision community addresses the broad field of human activity modeling by developing specialized methods for certain sub-tasks. Among others, video-based human activity modeling is concerned with a number of tasks such as inferring current and future semantic labels, predicting future continuous observations as well as imagining possible future label and feature sequences. All of these sub-tasks are important for human activity understanding. However, as they are usually independently, it is cumbersome and computationally expensive to employ them in real-time applications that require holistic activty understanding, such as human-robot interaction.

In this paper we present a semi-supervised probabilistic deep latent variable model that can represent both discrete labels and continuous observations as well as latent dynamics over time. This allows the model to solve several tasks at once without explicit fine-tuning. We focus here on the tasks of action classification, detection, prediction and anticipation as well as motion prediction and synthesis based on 3D human activity data recorded with Kinect.

The model combines the ideas of several variational autoencoder structures. Through the semi-supervised formulation, it can be trained based on partially labeled data and propagate semantic information over long periods of time. We further extend the model to capture hierarchical label structure and to model the dependencies between multiple entities, such as a human and objects.

Our experiments demonstrate that our principled approach to human activity modeling can be used to detect current and anticipate future semantic labels and to predict and synthesize future label and feature sequences. When comparing our model to state-of-the-art approaches, which are specifically designed for e.g. action classification, we find that our probabilistic formulation outperforms or is comparable to these task specific models.

Partly published at Human-robot cooperation and collaboration in manipulation: advancements and challenges, IROS workshop, 2018 and Precognition: Seeing through the Future, CVPR workshop, 2019

Complete list of publications

This section lists all work that was published during the doctoral studies 2015-2019.

2019

Bütepage, J. and Kjellström, H. and Kragic, D. (2019), Predicting the What and How - a Probabilistic Semi-Supervised Approach to Multi-Task Human Activity Modeling, *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*

Bütepage, J. and Poklukar, P. and Kragic, D. (2019), Modeling assumptions and evaluation schemes: On the assessment of deep latent variable models, *Uncertainty and Robustness in Deep Visual Learning Workshop, CVPR,* and *Robustness and Uncertainty Estimation in Deep Learning Workshop, ICML, 2019*

Bütepage, J. and Cruciani, S. and Kokic, M. and Welle, M. and Kragic, D. (2019), From Visual Understanding to Complex Object Manipulation, *Annual Review of Control, Robotics, and Autonomous Systems, volume 2, number 1, 2019*

2018

Bütepage, J. and He, J. and Zhang, C. and Sigal, L. and Mori, G. and Mandt, S. (2018), Informed Priors for Deep Representation Learning, *Symposium on Advances in Approximate Bayesian Inference, NIPS, 2018*

Bütepage, J. and Kjellström, H. and Kragic, D. (2018), Robustness and Uncertainty Estimation in Deep Learning, *arXiv preprint arXiv:1809.08875, 2018*

Bütepage, J. and Kragic, D. (2018), Detect, anticipate and generate: Semi-supervised recurrent latent variable models for human activity modeling, *presented at Human-robot cooperation and collaboration in manipulation: advancements and challenges, IROS workshop, 2018*

Bütepage, J. and Kjellström, H. and Kragic, D. (2018), Anticipating many futures: Online human motion prediction and synthesis for human-robot collaboration, *IEEE International Conference on Robotics and Automation (ICRA), 2018*

Zhang, C. and Bütepage, J. and Kjellström, H. and Mandt, S. (2018), Advances in variational inference, *IEEE Transactions on Pattern Analysis and Machine Intelligence*

2017

Bütepage, J. and Kragic, D. (2017), Human-Robot Collaboration: From Psychology to Social Robotics, *arXiv preprint arXiv:1705.10146*

Bütepage, J. and Black, M. and Kragic, D. and Kjellström, H. (2017), Deep representation learning for human motion prediction and classification, *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017*

2016

Vesper, C. and Abramova, E. and Bütepage, J. and Ciardo, F. and Crossey, B. and Effenberg, A. and Hristova, D. and Karlinsky, A. and McEllin, L. and Nijssen, S. and others (2016), Joint Action: Mental Representations, Shared Information and General Mechanisms for Coordinating with Others, *Frontiers in Psychology*, 7, 2039

Ghadirzadeh, A. and Bütepage, J. and Kragic, D. and Björkman, M. (2016), A sensorimotor reinforcement learning framework for physical Human-Robot Interaction, *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*(pp. 2682–2688). IEEE.

Bütepage, J. and Kjellström, H. and Kragic, D. (2016), Social Affordance Tracking over Time-A Sensorimotor Account of False-Belief Tasks, *Proc. 38th Annual Meeting of the Cognitive Science Society (CogSci)*(pp. 1014–1019). Cognitive Science Society.

Ghadirzadeh, A. and Bütepage, J. and Kragic, D. and Björkman, M. (2016), Self-learning and adaptation in a sensorimotor framework, *IEEE International Conference on Robotics and Automation (ICRA)*(pp. 551–558). IEEE.

Bibliography

[1] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 3rd ed. Pearson Education, 2003.

[2] R. Brooks, "New approaches to robotics," *Science*, vol. 253, no. 5025, pp. 1227–1232, 1991.

[3] J. K. O'Regan and A. Noë, "A sensorimotor account of vision and visual consciousness," *Behavioral and brain sciences*, vol. 24, no. 05, pp. 939–973, 2001.

[4] N. Silberman and R. Fergus, "Indoor scene segmentation using a structured light sensor," in *Proceedings of the International Conference on Computer Vision - Workshop on 3D Representation and Recognition*, 2011.

[5] M. Jeannerod, "Action monitoring and forward control of movements," in *Michael Arbib (Ed.), The Handbook of Brain Theory and Neural Networks. Second Edition.* MIT Press, 2003, p. 83â85.

[6] L. Montesano, M. Lopes, A. Bernardino, and J. Santos-Victor, "Learning object affordances: From sensory–motor coordination to imitation," *Robotics, IEEE Transactions on*, vol. 24, no. 1, pp. 15–26, 2008.

[7] C. Vesper, E. Abramova, J. Bütepage, F. Ciardo, B. Crossey, A. Effenberg, D. Hristova, A. Karlinsky, L. McEllin, S. R. Nijssen *et al.*, "Joint action: mental representations, shared information and general mechanisms for coordinating with others," *Frontiers in Psychology*, vol. 7, p. 2039, 2017.

[8] G. Rizzolatti and L. Craighero, "The mirror-neuron system," *Annu. Rev. Neurosci.*, vol. 27, pp. 169–192, 2004.

[9] J. Bütepage and D. Kragic, "Human-robot collaboration: From psychology to social robotics," *arXiv preprint arXiv:1705.10146*, 2017.

[10] C. E. Rasmussen, "Gaussian processes in machine learning," in *Summer School on Machine Learning*. Springer, 2003, pp. 63–71.

[11] R. Frigola-Alcalde, "Bayesian time series learning with gaussian processes," Ph.D. dissertation.

[12] C. Zhang, J. Butepage, H. Kjellstrom, and S. Mandt, "Advances in variational inference," *IEEE transactions on pattern analysis and machine intelligence*, 2018.

[13] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *International Conference on Learning Representations (ICLR)*, 2015.

[14] D. J. Rezende, S. Mohamed, and D. Wierstra, "Stochastic backpropagation and approximate inference in deep generative models," in *International Conference on Machine Learning*, 2014, pp. 1278–1286.

[15] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[16] A. Van Den Oord, N. Kalchbrenner, and K. Kavukcuoglu, "Pixel recurrent neural networks," in *Proceedings of the 33rd International Conference on International Conference on Machine Learning-Volume 48*. JMLR. org, 2016, pp. 1747–1756.

[17] L. Dinh, D. Krueger, and Y. Bengio, "Nice: Non-linear independent components estimation," *ICLR Workshop Track*, 2015.

[18] L. Dinh, J. Sohl-Dickstein, and S. Bengio, "Density estimation using real nvp," *International Conference on Learning Representations (ICLR)*, 2017.

[19] A. Ghadirzadeh, G. Kootstra, A. Maki, and M. Bjorkman, "Learning visual forward models to compensate for self-induced image motion," in *Robot and Human Interactive Communication, 2014 RO-MAN: The 23rd IEEE International Symposium on*. IEEE, 2014, pp. 1110–1115.

[20] G. Arvanitidis, L. K. Hansen, and S. Hauberg, "Latent space oddity: on the curvature of deep generative models," *ICLR*, 2018.

[21] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *international conference on machine learning*, 2016, pp. 1050–1059.

[22] C. L. Nehaniv and K. E. Dautenhahn, *Imitation and social learning in robots, humans and animals: behavioural, social and communicative dimensions.* Cambridge University Press, 2007.

[23] C. L. Nehaniv and K. Dautenhahn, "The correspondence problem," in *Imitation in animals and artifacts*, 2002, pp. 41–61.

[24] A. P. Shon, K. Grochow, and R. P. N. Rao, "Robotic imitation from human motion capture using gaussian processes," in *Humanoid Robots, 2005 5th IEEE-RAS*. IEEE, 2005, pp. 129–134.

[25] J. Mainprice and D. Berenson, "Human-robot collaborative manipulation planning using early prediction of human motion," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 299–306.

[26] L. Sartori, C. Becchio, B. G. Bara, and U. Castiello, "Does the intention to communicate affect action kinematics?" *Consciousness and cognition*, vol. 18, no. 3, pp. 766–772, 2009.

[27] L. M. Sacheli, E. Tidoni, E. F. Pavone, S. M. Aglioti, and M. Candidi, "Kinematics fingerprints of leader and follower role-taking during cooperative joint actions," *Experimental brain research*, vol. 226, no. 4, pp. 473–486, 2013.

[28] C. Scorolli, M. Miatton, L. A. Wheaton, and A. M. Borghi, "I give you a cup, i get a cup: a kinematic study on social intention," *Neuropsychologia*, vol. 57, pp. 196–204, 2014.

[29] A. D. Dragan, K. C. Lee, and S. S. Srinivasa, "Legibility and predictability of robot motion," in *2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 2013, pp. 301–308.

[30] H. S. Koppula, A. Jain, and A. Saxena, "Anticipatory planning for human-robot teams," in *Experimental Robotics*. Springer, 2016, pp. 453–470.

[31] H. B. Amor, G. Neumann, S. Kamthe, O. Kroemer, and J. Peters, "Interaction primitives for human-robot cooperation tasks," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014, pp. 2831–2837.

[32] J. Bütepage, H. Kjellström, and D. Kragic, "Anticipating many futures: Online human motion prediction and synthesis for human-robot collaboration," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, 2018.

Part II

Included Publications