

# Perceive, Represent, Generate: Translating Multimodal Information to Robotic Motion Trajectories

Fábio Vital

Instituto Superior Técnico, Universidade de Lisboa

Lisboa, Portugal

fabioital@tecnico.ulisboa.pt

## ABSTRACT

In this thesis, we contribute a novel pipeline that maps perceptual information of different modalities (e.g., visual or sound), corresponding to a sequence of commands, to an adequate sequence of movements to be executed by a robot. Our *Perceive-Represent-Generate* (PRG) framework comprises three stages. In the first stage, we perceive and pre-process the given inputs, isolating individual commands. The second stage, a core element in our pipeline, uses a deep generative model that captures the joint distribution of the perceptual information and the robot movement. Such representation enables the robot to determine the adequate movement given the perceptual input from a command. Finally, the third stage combines the movement for the different individual commands into a dynamic movement primitive, which the robot must execute. We evaluate our pipeline in the context of robotic handwriting, where the robot receives as input a word or sentence (in printed form, handwritten form, or as a sound stream) and determines the complete movement required to write it. We evaluate each stage of our pipeline separately, and as a whole. We discuss the performance of different multimodal generative models within our PRG framework and show that our pipeline can generate coherent and readable handwritten words, regardless of the modalities provided to the model.

## KEYWORDS

Robotic Handwriting, Dynamic Movement Primitives (DMPs), Multimodal Learning

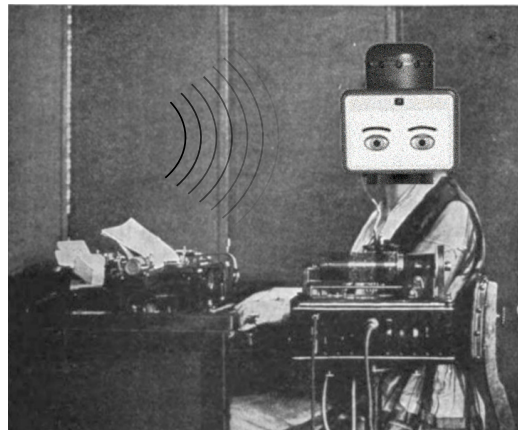
### ACM Reference Format:

Fábio Vital. 2022. Perceive, Represent, Generate: Translating Multimodal Information to Robotic Motion Trajectories. In *Proc. of the 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2022)*, Auckland, New Zealand, May 9–13, 2022, IFAAMAS, 9 pages.

## 1 INTRODUCTION

Recent advancements in artificial perception [18, 21] and actuation [2, 12] have fostered the widespread use of robotic platforms in a myriad of tasks, from autonomous driving [3, 31] to industrial manufacturing [28] as well as in medical [5, 19, 22] or education [6, 15] scenarios. Furthermore, robots are increasingly expected to perform tasks in collaboration with humans, raising significant challenges regarding the quality of their interaction and the mismatch between their perceptual, cognitive, and actuation capabilities [14, 26].

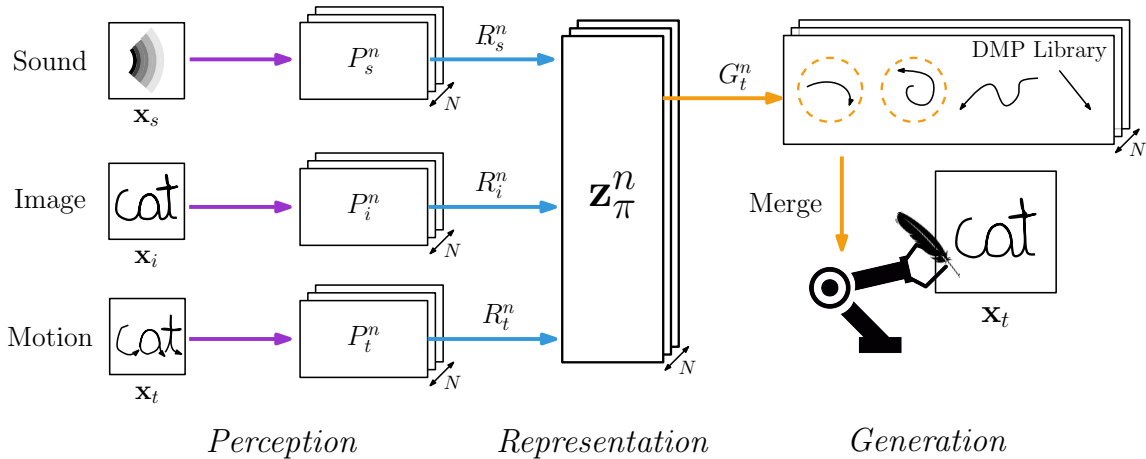
Proc. of the 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2022), P. Faliszewski, V. Mascardi, C. Pelachaud, M.E. Taylor (eds.), May 9–13, 2022, Auckland, New Zealand. © 2022 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). All rights reserved.



**Figure 1:** In this work, we study how to translate multimodal information into a sequence of movements executed by a robotic agent. In the illustration, a human can dictate a text or present it visually. The agent must process and decompose that information to derive the movement to type each text letter. Adapted from *The Expert Typist* by Charles Smith (1922).

In such scenarios, humans are expected to employ different natural communication channels to provide instructions to the robotic platform, such as speech and motion. In this work, we address the problem of *how to translate multimodal commands*, provided by a human user through different communication channels, to a *sequence of movements* executed by a robotic agent. In particular, we consider a scenario where the human user provides high-dimensional perceptual data (e.g., sound, images, motion trajectories) related to the agent’s task, and the agent’s role is to decompose the raw observations into sequential individual actions to perform. Consider the example of a *robotic dictaphone*, as depicted in Fig. 1: a human user dictates the text to be written, and the robot must be able to decompose the provided words to perform the trajectory associated with each individual letter. Moreover, at any given moment, the robot’s performance must be robust to the fact that the human might not employ all communication channels to provide task-related information.

In this work, we contribute with a novel three-stage framework *Perceive-Represent-Generate* (PRG) that maps multimodal commands, provided from raw perceptual data generated by a human user, to a movement to be executed by the robot. Initially, the agent *perceives* the environment, collecting and processing the raw multimodal observations into a sequence of individual task components



**Figure 2: The proposed “Perceive-Represent-Generate” (PRG) pipeline for multimodal actuation, instantiated for the robotic Dictaphone scenario. In the *Perception* phase, the model collects commands from the human user through multimodal information  $\mathcal{X} = \{x_i, x_s, x_t\}$  and performs an initial preprocessing to efficiently compress raw observation data and isolate individual commands  $P_m^n, n \in [0, N]$ , using the perception maps  $\mathcal{P}$ . In the *Representation* phase, we employ the representation maps  $\mathcal{R}$ , provided by the multimodal generative model, to encode the individual commands  $P_m^n$  in a latent representation  $z_\pi^n$ , suitable to be encoded from partial observations. In the *Generation* phase the model decodes each latent representation using the target generation map  $G_t^n$  and merges the individual motion information to generate target trajectory data  $x_t$ .**

(e.g., letters in a word). Subsequently, the agent *represents* the individual task components, mapping them into multimodal representations. Crucially, as humans may not employ all communication channels to provide information to the agent, such multimodal representation must be robust to missing modality information. Finally, in the third stage, the agent *generates* and merges the motion information provided by the individual representations in order to execute the estimated action/motion.

We instantiate our PRG pipeline in the scenario of the robotic Dictaphone, where the agent is provided with textual information (through a combination of sound, image, or motion trajectory observations) and generates a single motion trajectory related to the target word mimicking human handwriting. We conduct a quantitative evaluation of the representation stage separately, giving us insight into how well we encode the multimodal perceptions and thereby checking the degree of effectiveness that the agent can cross generate missing modalities. We also evaluate our approach qualitatively, focusing on how well the agent can generate smooth handwritten words approximating it to the calligraphy of a human. The results show that our approach can robustly map multimodal commands to generate accurate handwritten word samples, regardless of the set of communication channels employed by the human to provide the commands.

In summary, the main contributions of this work are two-fold:

- We propose a novel three-stage pipeline *PRG* that allows translating multimodal information provided by a human user to an adequate movement to be executed by a robot. Crucially, such mapping is *robust* to missing modality information, as the human may not always provide information through all available communication channels;

- We instantiate our PRG approach in a novel Robotic Dictaphone scenario where textual information is converted to robotic motion trajectory, mimicking human handwriting. Our results show that, regardless of the communication channel employed by the human user (e.g., speech, image), our pipeline can accurately translate such information to generate coherent handwritten samples.

## 2 BACKGROUND

This section describes two topics that we use in this research work, namely variational autoencoders and dynamic movement primitives.

### 2.1 Variational Autoencoder

The variational autoencoder (VAE) model is often used to learn a low-level representation of single-modality perceptual data. In such a scenario, we consider that the environment provides information  $\mathbf{x}$ , encoded in a low-dimensional latent variable  $\mathbf{z}$ . Formally, training a VAE model amounts to estimating the lower-bound of the evidence  $p(\mathbf{x})$ ,  $\mathcal{L}_{\text{VAE}}$ , resorting to a variational approach,

$$\mathcal{L}_{\text{VAE}} = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} [\log p_\theta(\mathbf{x}|\mathbf{z})] - \mathbb{KL}(q_\phi(\mathbf{z}|\mathbf{x}) \parallel p(\mathbf{z})), \quad (1)$$

where  $p(\mathbf{z})$  is a pre-specified prior, often a unitary Gaussian distribution  $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ ,  $p_\theta(\mathbf{x}|\mathbf{z})$  is the likelihood distribution (the “decoder”), parameterized by  $\theta$  and  $q_\phi(\mathbf{z}|\mathbf{x})$  is the proposal distribution (the “encoder”), parameterized by  $\phi$ .

The variational framework has been extended to consider the conditional generation of input data  $\mathbf{x}$  over output data (labels)  $\mathbf{y}$ . The Conditional VAE (CVAE) model [24] attempts to model the

conditional likelihood  $p(\mathbf{x} | \mathbf{y})$ , following the conditional ELBO:

$$\begin{aligned} \mathcal{L}_{\text{CVAE}}(\mathbf{x}, \mathbf{y}) = & \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}, \mathbf{y})} [\log p_\theta(\mathbf{x}|\mathbf{y}, \mathbf{z})] \\ & - \mathbb{KL}(q_\phi(\mathbf{z}|\mathbf{x}, \mathbf{y}) \parallel p(\mathbf{z})) \end{aligned} \quad (2)$$

VAE models have also been properly extended to scenarios with multiple modalities. The Associative Variational Autoencoder (AAVE) model [30] is able to learn a common latent representation of two modalities  $\mathbf{x}_1, \mathbf{x}_2$ , following the multimodal ELBO:

$$\begin{aligned} \mathcal{L}_{\text{AAVE}}(\mathbf{x}_1, \mathbf{x}_2) = & \mathcal{L}_{\text{VAE}}(\mathbf{x}_1) + \mathcal{L}_{\text{VAE}}(\mathbf{x}_2) \\ & - \alpha \mathbb{KL}^*(q_\phi(\mathbf{z}_{\mathbf{x}_1}|\mathbf{x}_1) \parallel q_\phi(\mathbf{z}_{\mathbf{x}_2}|\mathbf{x}_2)) \end{aligned} \quad (3)$$

where  $\mathbb{KL}^*(p \parallel q)$  is the symmetrical Kullback-Leibler between two distributions  $p$  and  $q$ , and  $\alpha$  is the parameter that weights the similarity between the modality-specific latent spaces during training. Another variational autoencoder-based framework have been recently proposed to allow learning a multimodal representation, robust to missing modality information, scalable to more than two modalities. The Multimodal Unsupervised Sensing (MUSE) model [27] employs hierarchical representation levels (modality-specific and multimodal), following a loss function:

$$\begin{aligned} \mathcal{L}_{\text{MUSE}}(\mathbf{x}_1, \mathbf{x}_2) = & \mathcal{L}_{\text{VAE}}(\mathbf{x}_1) + \mathcal{L}_{\text{VAE}}(\mathbf{x}_2) \\ & + \mathbb{E}_{q_\phi(\mathbf{c}_{1:2}|\mathbf{x}_{1:2})} \left( \beta \mathbb{KL}(q_\phi(\mathbf{z}_\pi | \mathbf{c}_1, \mathbf{c}_2) \parallel p(\mathbf{z}_\pi)) \right. \\ & - \mathbb{E}_{q_\phi(\mathbf{z}_\pi|\mathbf{c}_1)} [\gamma_1 \log p_\theta(\mathbf{c}_1 | \mathbf{z}_\pi)] \\ & \left. - \mathbb{E}_{q_\phi(\mathbf{z}_\pi|\mathbf{c}_2)} [\gamma_2 \log p_\theta(\mathbf{c}_2 | \mathbf{z}_\pi)] \right) \end{aligned} \quad (4)$$

where  $\mathbf{c}_m \sim q_\phi(\mathbf{z}_m | \mathbf{x}_m)$  are low-dimensional, modality-specific codes, sampled from the low-level distributions  $q_\phi(\mathbf{z}_m | \mathbf{x}_m)$ .

## 2.2 Dynamic Movement Primitives

Dynamic Movement Primitive (DMP) [9] is a method to learn and control trajectories. The original DMP formulation that we use was designed for discrete movements, having start and endpoints. Two systems define a DMP: the first one, called *transformation system*, which can be represented as a second-order dynamical system

$$\tau \dot{z} = \alpha_z (\beta_z (g - y) - z) + f, \quad (5)$$

$$\tau \dot{y} = z, \quad (6)$$

where  $y, \dot{y}$ , and  $\dot{z}$  are the position, velocity and scaled acceleration, respectively.  $\tau$  is a time constant, and  $\alpha_z$  and  $\beta_z$  are positive constants.  $g$  is the desired endpoint (the goal) and the point attractor of the system.

The second system, called the *canonical system*, is a first-order linear dynamical system

$$\tau \dot{x} = -\alpha_x x, \quad (7)$$

where  $\alpha_x$  is a constant. For a randomly chosen initial state,  $x_0 = 1$  describes the start of time evolution. As  $x$  converges to 0, monotonically, the trajectory approaches its final state. Thus,  $x$  serves as a phase signal, and  $x = 0$  is a stable fixed point of (7). Consequently, we can write  $f$ , in (5), as:

$$f(x, y, g) = \frac{\sum_{i=1}^N \Psi_i(x) w_i}{\sum_{i=1}^N \Psi_i(x)} x(g - y_0) \quad (8)$$

where each  $\Psi_i(x)$  is a Gaussian basis function:

$$\Psi_i(x) = \exp\left(-\frac{1}{2\sigma_i^2} (x - c_i)^2\right), \quad (9)$$

with  $c_i$  and  $\sigma_i$  being constants corresponding to the center and width of the basis function  $\Psi_i$ , respectively.  $y_0$  is the trajectory's initial state,  $y_0 = y(0)$ .

Such a definition of  $f$  starts to vanish when the generated trajectory approaches  $g$ , forcing the global equilibrium point (point attractor) to appear at  $(z, y, x) = (0, g, 0)$ . In addition, we assume that  $g \neq y_0$ , meaning the offset between the end and start point of the trajectory is never 0. To generate complex trajectories, we can use learning algorithms, such as locally weighted regression, to regulate the parameters  $w_i$  of (8), forming discrete pattern generators as  $y$  tends to  $g$ .

## 3 METHODOLOGY

In this work, we consider the problem of translating multimodal information provided by a human user into motion trajectories, suitable to be executed by a robotic agent. We propose a novel three-stage pipeline *Perceive-Represent-Generate* (PRG) to address such a problem, as shown in Fig. 2. In the first stage (*Perception*), the agent collects multimodal information provided by the human user and, if required, processes the raw observation data. In the second stage (*Representation*), the agent encodes the available information in a multimodal latent representation  $\mathbf{z}_\pi$ . Finally, in the third stage (*Generation*), the agent decodes the representation to generate the target modality suitable for the actuation of the robot. In the following sections, we discuss in detail each of these stages.

### 3.1 Perception

In the initial stage of our pipeline (*Perception*), the agent collects the commands from the human user, i.e., directly from the raw multimodal observations provided by the user. In this work, we assume that the human provides commands regarding the agent's task through  $M$  different communication channels,  $\mathcal{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_M\}$ , where  $\mathbf{x}_m \in \mathbb{R}^m$  is the information provided by an input "channel". Each modality may correspond to a different type of information (e.g., image, sound, motion trajectory) acquired by the agent's sensors.

A provided command  $\mathbf{x}_m$  can be decomposed into a list of individual commands  $\{\mathbf{x}_m^1, \mathbf{x}_m^2, \dots, \mathbf{x}_m^N\}$ , to be performed sequentially by the agent (e.g., the sound of a word can be decomposed into individual letters to be written). As such, the perception module decomposes the raw observations into individual observations  $\mathbf{x}_m^n, n \in \{1, 2, \dots, N\}$ . In particular, we employ a set of perception maps  $\mathcal{P} = \{P_1, P_2, \dots, P_M\}$ , where each function  $P_m : \mathbf{x}_m \rightarrow \mathbf{x}_m^n$  maps the original observation  $\mathbf{x}_m$  to a set of *individual* representations  $\mathbf{x}_m^1, \mathbf{x}_m^2, \dots, \mathbf{x}_m^N$ .

The proposed pipeline is agnostic to the nature and number of the perception modules  $\mathcal{P}$  because they can be learned or heuristically defined. For instance, to process sound information, one can employ a generative model to encode a low-dimensional representation of available models (such as wav2vec 2.0 [4]) or employ a pre-trained speech-to-text model to retrieve semantic information and decompose the words into individual letters.

### 3.2 Representation

In the *Representation* stage, we encode the individual modality-specific information  $\mathbf{x}_m^n$  in a corresponding multimodal latent representation  $\mathbf{z}_\pi^n \in \mathcal{Z}$ . In particular, we intend to learn a set of latent representation maps  $\mathcal{R} = \{R_1, R_2, \dots, R_L\}$ , where each map  $R_l$  takes the form  $R_l : \text{proj}_l \rightarrow \mathcal{Z}$ , where  $\mathcal{Z}$  is the multimodal latent space and  $\text{proj}_l$  projects the input space  $\mathcal{X}$  to a subspace of  $K$  modalities,  $\mathcal{X}_l = \{\mathbf{x}_{l_1}, \mathbf{x}_{l_2}, \dots, \mathbf{x}_{l_K}\}$ . We note that the number of representation maps can be larger than the number of modalities, for example by accounting for the combination of different input modalities.

To learn such mappings in an unsupervised learning framework, we instantiate and train a multimodal VAE (MVAE) model. The encoders of the MVAE model correspond to the representation maps  $\mathcal{R}$  and the decoders of the MVAE model correspond to the inverse representation maps  $\mathcal{G} = \{G_1, G_2, \dots, G_L\}$ , that allow for the generation of modality-specific information and cross-modality inference. Finally, we note that our pipeline is agnostic to the nature of the MVAE model instantiated at this stage.

### 3.3 Generation

In the final stage of the pipeline (*Generation*), the agent generates motion trajectory data  $\mathbf{x}_t$  from the sequence of multimodal representations  $\{\mathbf{z}_\pi^0, \dots, \mathbf{z}_\pi^N\}$ . In particular, we re-purpose the inverse representation maps  $\mathcal{G}$ , previously trained, and select the motion trajectory generation map  $G_t : \mathcal{Z} \rightarrow \mathbf{x}_t$  to generate each individual target motion. Each target motion can then be processed, accommodating to eventual task-related restrictions. This is done using another perception map  $P'_t$ , where  $P'_t : \mathbf{x}_t^n \rightarrow \mathbf{x}_t^{n'}$  and  $n \in \{1, 2, \dots, N\}$ , e.g. translating the next trajectory  $\mathbf{x}_t^{n+1}$  to match the final position of the previous motion  $\mathbf{x}_t^n$  or scaling the trajectories. After processing all motions, we concatenate and transform them into a single DMP, ready to be executed by the agent.

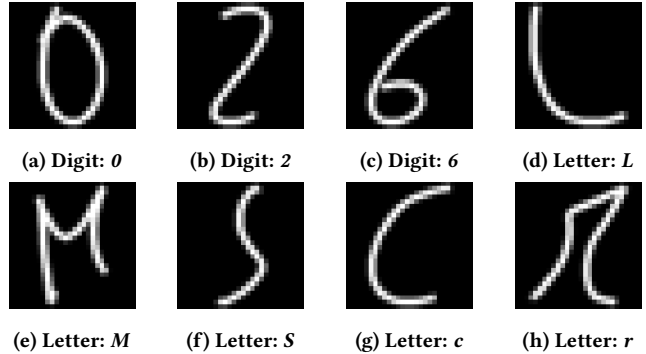
An additional post-processing module may be required for the agent to act upon its environment effectively. For example, in the case of a robotic handwriting task, generated motion information might be unsuitable for a given robotic platform and require specific modification (e.g., transformation to joint-angle information) before execution. Our approach is agnostic to the nature and number of post-processing modules.

## 4 EVALUATION

In this section, we present an instantiation of our proposed framework in the context of the *Robotic Dictaphone* scenario, where the goal of the robot is to generate handwritten word samples from information provided by the human user through speech. We conduct a quantitative and qualitative evaluation of the generative capability of our model regarding its capability to map observations to a latent representation and show that our approach is robust to missing modality information. Finally, we present qualitative samples from handwritten words generated from our pipeline, showcasing the effectiveness of our complete pipeline.

### 4.1 Scenario

In this work, we consider the following scenario of a *Robotic Dictaphone*. A human provides a word to be transcribed by the robot



**Figure 3: Image samples of handwritten digits and letters randomly retrieved from the extended “UJI Char Pen 2” dataset employed to train the representation maps  $\mathcal{R}$ .**

through different communication channels  $\mathcal{X} = \{\mathbf{x}_s, \mathbf{x}_i, \mathbf{x}_t\}$ . Specifically,  $\mathbf{x}_s$  is the sound corresponding to the word,  $\mathbf{x}_i$  is a sequence of images, where the  $i$ th image is a handwritten representation of the  $i$ th letter of the same word, and  $\mathbf{x}_t$  is a sequence of 2D motions, where the  $i$ th motion is a handwritten representation of the  $i$ th letter of the corresponding word.

The agent’s goal is to map such multimodal information in an internal representation, suitable for the downstream generation of coherent samples of the handwritten words. However, despite being available, the human user may only employ a subset of such communication channels to provide the goal words. As such, the agent must learn to encode a representation that is robust to partial observations. We now instantiate our pipeline for the case of the Robotic Dictaphone scenario, describing in detail each of the modules.

**4.1.1 Perception.** In this first stage, the raw observation data provided by the human user is processed and decomposed into individual “commands”, in this case individual letters in the word. To do so, we define the perception maps specific to each communication channel  $\mathcal{P} = \{P_s, P_i, P_t\}$ . For the sound perception map  $P_s$ , we employ *wav2vec 2.0*, a self-supervised learning framework for speech recognition [4]. As such, we process the raw audio data into the label information associated with each letter, allowing for more efficient downstream representation. For the imaging modality, we utilize a function map,  $P_i$ , that unpacks a given input sequence.  $\mathbf{x}_i$  is a sequence of image tensors and, when given to  $P_i$ , we get  $\mathbf{x}_i^n$ , where  $n \in \{1, 2, \dots, N\}$ ,  $N$  is the number of letters of the underlying word, and  $\mathbf{x}_i^n$  is then the image tensor that corresponds to the  $n$ -th letter of the underlying word. For  $P_t$ , we also unpack the sequence of letter trajectories and then normalize it, given a pre-computed mean and variance from the training dataset (used to train each multimodal VAE employed in the next stage).

**4.1.2 Representation.** In this stage, the observations corresponding to the individual “commands” are mapped to the multimodal latent space  $\mathbf{z}_\pi$ . We employ and compare different multimodal variational autoencoder models to learn such representation due to their low memory requirements and stable training. In particular,

**Table 1: Standard log-likelihood metrics for the Representation models  $\mathcal{R}$  in the “UJI Char Pen 2” dataset, estimated resorting to 1000 importance weighted samples.**

(a) $R_{CVAE}(x_s, x_t)$		(b) $R_{AAVE}(x_s, x_t)$				(c) $R_{AAVE}(x_i, x_t)$			
$\log p(x_t x_s)$		$\log p(x_t)$	$\log p(x_s)$	$\log p(x_t x_s)$	$\log p(x_s x_t)$	$\log p(x_t)$	$\log p(x_i)$	$\log p(x_t x_i)$	$\log p(x_i x_t)$
-192.75		-197.55	-4.17	-189.28	1.94	-197.69	-743.57	-186.28	-730.44

(d) $R_{MUSE}(x_s, x_i, x_t)$						
$\log p(x_t)$	$\log p(x_s)$	$\log p(x_i)$	$\log p(x_t x_s)$	$\log p(x_s x_t)$	$\log p(x_t x_i)$	$\log p(x_i x_t)$
-198.04	-4.53	-742.49	-198.10	1.96	-193.63	-735.02

we instantiate four different representation models, each able to account for a different number of modalities:

- $R_{CVAE}(x_s, x_t)$ , the CVAE model [24] able to generate trajectory information  $x_t$  conditioned on sound information  $x_s$ . The model is trained resorting to the loss function of Eq. (2).
- $R_{AAVE}(x_s, x_t)$ , the AVAE model that learns a joint representation of trajectory information  $x_t$  and sound information  $x_s$ . The model is trained resorting to the loss function of Eq. (3).
- $R_{AAVE}(x_i, x_t)$ , the AVAE model that learns a joint representation of trajectory information  $x_t$  and image information  $x_i$ . The model is trained resorting to the loss function of Eq. (3).
- $R_{MUSE}(x_s, x_t, x_i)$ , the MUSE model, able to learn a joint representation from sound  $x_s$ , trajectory  $x_t$  and image information  $x_i$ . The model is trained resorting to the loss function of Eq. (4).

All the representation models are previously trained on data provided from the “UJI Char Pen 2” dataset, from which only one-stroke-formed digits and letters are processed [16]. To address the small number of samples presented in that dataset, we learn a probabilistic model of each character and re-sample with perturbations constrained in a kinematics feature space, following the procedure described in [30]. This way, we generate around 60,000 samples of  $28 \times 28$  grayscale images and 200-dimensional representations of the associated trajectories, corresponding to 62 different classes (uppercase and lowercase English letters and all digits). In addition, we normalize all trajectories to the unit interval. We present samples from the extended dataset in Fig. 3.

**4.1.3 Generation.** In this final stage, the collected representations are decoded into individual motion trajectories. In the robotic Dictaphone scenario, we apply several processing steps before transforming each trajectory into a DMP, through the final perceptual map  $P'_t$ :

- (1) We define the height of each letter trajectory so that it has the expected proportion in the final word. Heuristically, we define that every uppercase and lowercase letters have 2 “boxes” of height, except for the lowercase letters  $\{a, c, e, i, m, n, o, r, s, u, v, w, x, z\}$  that only have one box of height.
- (2) We also translate vertically each letter trajectory. Heuristically, every letter must start at the origin,  $y = 0$ , except for

the letters  $\{f, g, j, p, q, y\}$  which start at  $y = -b$ , where  $b$  is the height of the box computed in (1).

- (3) We define a fixed horizontal distance that will be used to separate any two consecutive trajectories.
- (4) In the final processing step, we use a custom algorithm to exploit the space between each letter trajectory to derive a partial motion to connect the two letters. We chose this approach over using an algorithm that merges movements following a straight line. Our algorithm, instead, computes such intermediate trajectory iteratively, considering several aspects in each increment, resulting in a more natural and smooth handwriting motion, see Algorithm 1 for more details.

After all processing steps, the model generates  $n$  letter trajectories plus  $n - 1$  connection trajectories. These steps are necessary to generate handwritten samples coherent with human ones. The final step in the pipeline is to concatenate all the letter and connection trajectories and convert them into a single DMP: the  $i$ -th connection trajectory appears between the  $i$ -th and  $(i + 1)$ -th letter trajectories. In the end, we have one DMP, suitable for the robot’s actuation to write the target word.

## 4.2 Results

In this section, we present the results of the evaluation of our pipeline, regarding the performance of the representation module (Section 4.2.1) and the handwritten samples generated by the complete pipeline (Section 4.2.2). We present quantitative and qualitative results to attest to the following: (i) our pipeline allows the effective representation of multimodal information, being robust to missing information; (ii) our pipeline allows the generation of coherent and high-quality samples of handwritten words.

**4.2.1 Representation.** We evaluate quantitatively the generative performance of the representation models employed in our pipeline. We present standard *log-likelihood* metrics regarding the marginal and conditional log-likelihoods, that are estimated resorting to 1000 importance-weighted samples. We present the results in Table 1. In addition, we present two metrics that consider the cross-modality generation performance of the representation, as proposed in [23]: *accuracy* and *modality-distance*. The former evaluates if the samples generated by cross-modality inference are semantically coherent with the available modality data that was provided to the model,

**Table 2: Accuracy (%) metrics for the Representation models  $\mathcal{R}$  in the “UJI Char Pen 2” dataset.**

(a) $R_{CVAE}(x_s, x_t)$		(b) $R_{AAVE}(x_s, x_t)$			(c) $R_{AAVE}(x_i, x_t)$			(d) $R_{MUSE}(x_s, x_i, x_t)$						
Target	input	Target	input		Target	input		Target	input					
	$x_s$		$x_t$	$x_s$		$x_t$	$x_i$		$x_t$	$x_s$	$x_i$	$x_t, x_s$	$x_t, x_i$	$x_s, x_i$
$x_t$	81.52	$x_t$	-	55.82	$x_t$	-	67.19	$x_t$	-	69.18	33.39	-	-	69.00
		$x_s$	62.96	-	$x_i$	64.03	-	$x_s$	52.09	-	34.94	60.81	-	-
								$x_i$	42.06	58.12	-	-	65.92	-

**Table 3: Modality-distance metrics for the Representation models  $\mathcal{R}$  in the “UJI Char Pen 2” dataset.**

(a) $R_{CVAE}(x_s, x_t)$		(b) $R_{AAVE}(x_s, x_t)$			(c) $R_{AAVE}(x_i, x_t)$			(d) $R_{MUSE}(x_s, x_i, x_t)$						
Target	input	Target	input		Target	input		Target	input					
	$x_s$		$x_t$	$x_s$		$x_t$	$x_i$		$x_t$	$x_s$	$x_i$	$x_t, x_s$	$x_t, x_i$	$x_s, x_i$
$x_t$	1.4395	$x_t$	4.7258		$x_t$	-	0.9678	$x_t$	-	4.6734	6.2185	-	-	2.8230
					$x_i$	0.0059	-	$x_i$	0.0458	0.1475	-	0.0797	-	

e.g. generated images from label “c” should be classified as image samples of that letter. The latter evaluates if the samples generated by cross-modality inference are similar to the original samples in the dataset, e.g. generated images from label “c” should account for the different ways an image of that letter can be handwritten. We show the results of such evaluation in Table 2 and Table 3.

The results reveal that there is a clear balance in performance between intrinsic performance of the representation models and scalability to higher number of modalities. Even though  $R_{CVAE}$  has a slightly worse conditional log-likelihood  $\log p(x_t|x_s)$  than  $R_{AAVE}(x_s, x_t)$ , see Table 1, the accuracy and modality-distance results (Tables 2 and 3, respectively) are far better than both models,  $R_{AAVE}(x_s, x_t)$  and  $R_{MUSE}$ . However, it is unable to learn a joint-representation from multiple modalities, nor scale to higher number of modalities. In contrast,  $R_{AAVE}(x_s, x_t)$ ,  $R_{AAVE}(x_i, x_t)$  and  $R_{MUSE}$  models are able to learn a quality joint-modality latent representation and map multimodal observations into a common latent space. Both  $R_{AAVE}$  models have better likelihood metrics than  $R_{MUSE}$ , except for  $\log p(x_s|x_t)$ , for  $R_{AAVE}(x_s, x_t)$ , and  $\log p(x_i|x_t)$ , for  $R_{AAVE}(x_i, x_t)$ , still, the results for this metric are very similar.  $R_{AAVE}(x_s, x_t)$  and  $R_{AAVE}(x_i, x_t)$  outperform  $R_{MUSE}$  in all modality-distance evaluations and in the accuracy metrics when giving only one modality as input, except when classifying  $x_t$  giving as input  $x_s$  (which could be to the fact that we are giving the same weight for the associative KL and individual VAEs,  $\alpha = 1$  in Eq. (3), for the  $R_{AAVE}(x_t, x_s)$  model). Still,  $R_{AAVE}$  models are unable to learn a multimodal latent space for more than two modalities without combinatorially exploding the number of parameters.  $R_{MUSE}$ , on the other hand, is able to compositionally account for the information provided by multiple modalities to encode a more robust latent representation, as shown in Table 2 from the increased, or maintaining similar, accuracy performance when the model is provided with more than one modality.

Furthermore, we can also evaluate the potential of the representations in encoding a representation robust to missing modality information by observing the trajectory samples generated from sound and image information. We show examples of such samples

in Table 4. We can see that the generated letter trajectories are quite different, depending on the representation model used. We can also see that the generated trajectories given the model  $R_{MUSE}(x_s, x_i, x_t)$  appear more wavy and unnatural than the other models. This could also be related with the interpretation we proposed above.

**4.2.2 Generation.** Finally, we qualitatively evaluate the efficacy of our full PRG framework in generating handwritten word samples  $x_t$ , from image  $x_i$  or sound  $x_s$  information. We provide samples of the handwritten words generated by the different representation models in Table 5. Once again we see that our PRG framework is able to generate high-quality and coherent handwritten word samples regardless of the representation model employed and of the modality available to the framework. In particular, we observe that the  $R_{CVAE}$  and  $R_{AAVE}$  representation models allow for the generation of more high-quality word samples, yet are unable to scale to more than two modalities. On the other hand,  $R_{MUSE}$  still allows for the generation of coherent word samples (despite a loss in quality in comparison with the other models), able to learn a multimodal representation robust to missing modality information, scalable to settings with larger number of modalities.

## 5 RELATED WORK

In this section, we present relevant literature regarding robotic control and two fundamental components of our approach: multimodal representation learning and dynamic motion control.

### 5.1 Human-Robot Interaction

In robotic control, there are several studies focused not only on how to control an agent through the use of commands, but also on how to simplify that communication. For instance, turning the set of commands more flexible and abstract to be able to interpret some level of uncertain information [20] turning instructions more high-level and, at the same time, giving more freedom to the agent.

There are several examples of works, in different sectors, that use a set of instructions (commands) to control a robotic agent: in robotic navigation through the use of directional voice instructions

---

**Algorithm 1:** Algorithm to create an intermediate (connection) trajectory between two consecutive trajectories.

---

**Input:**  $\mathbf{x}_T^i, \mathbf{x}_T^{i+1}$  - Motion trajectories;  
 $\delta$  - small trajectory increment;  
 $\alpha_{\max}$  - Maximum angular difference allowed between vectors;  
 $\theta_o$  - Penalization cost over new connection points based on the index of the point in  $\mathbf{x}_T^{i+1}$ ;  
 $\theta_a$  - Penalization cost over the angle between the last connection trajectory vector and the new candidate vector to be included in the connection trajectory;  
 $\theta_f$  - Penalization cost over the distance between new candidate point and the initial point of  $\mathbf{x}_T^{i+1}$ ;  
 $\theta_p$  - Penalization cost over the distance between new candidate point and the point in  $\mathbf{x}_T^{i+1}$  to connect (target).

**Result:** Trajectory,  $\mathbf{x}_T^i$ , connecting  $\mathbf{x}_T^i$  to  $\mathbf{x}_T^{i+1}$

```

 $\hat{x} \leftarrow \langle 1, 0 \rangle;$ 
 $N_i \leftarrow \text{length}(\mathbf{x}_T^i);$ 
 $N_{i+1} \leftarrow \text{length}(\mathbf{x}_T^{i+1});$ 
 $\bar{\mathbf{x}}_I \leftarrow [];$ 
 $c \leftarrow \mathbf{x}_T^i[N-1] - \mathbf{x}_T^i[N-2];$ 
 $\text{costs} \leftarrow [0, \dots, 0], \text{costs} \in \mathbb{R}^{N_{i+1}};$ 
 $\text{angles} \leftarrow [0, \dots, 0], \text{angles} \in \mathbb{R}^{N_{i+1}};$ 
 $\text{cand} \leftarrow [0, \dots, 0], \text{cand} \in \mathbb{R}^{N_{i+1}};$ 
do
  for  $j \leftarrow 0$  to  $N_{i+1} - 1$  do
     $\alpha \leftarrow \min(\alpha_{\max}, \angle(\mathbf{x}_T^{i+1}[j], \hat{x}) - \angle(c, \hat{x}));$ 
     $\text{angles}[j] \leftarrow \alpha;$ 
     $\text{cand}[j] \leftarrow c + \delta \cdot \langle \cos(\alpha), \sqrt{1 - \cos^2(\alpha)} \rangle;$ 
  end
  for  $j \leftarrow 0$  to  $N_{i+1} - 1$  do
     $\text{costs}[j] \leftarrow \theta_o \left( \frac{j}{N_{i+1}} \right) + \theta_a \left( \frac{\text{angles}[j] - \min(\text{angles})}{\max(\text{angles}) - \min(\text{angles})} \right);$ 
     $\text{costs}[j] \leftarrow \text{costs}[j] +$ 
     $\theta_f \left( \frac{\|\mathbf{x}_T^{i+1}[0] - \text{cand}[j]\| - \min_k (\|\mathbf{x}_T^{i+1}[0] - \text{cand}[k]\|)}{\max_k (\|\mathbf{x}_T^{i+1}[0] - \text{cand}[k]\|) - \min_k (\|\mathbf{x}_T^{i+1}[0] - \text{cand}[k]\|)} \right);$ 
     $\text{costs}[j] \leftarrow \text{costs}[j] +$ 
     $\theta_p \left( \frac{\|\mathbf{x}_T^{i+1}[j] - \text{cand}[j]\| - \min_k (\|\mathbf{x}_T^{i+1}[j] - \text{cand}[k]\|)}{\max_k (\|\mathbf{x}_T^{i+1}[j] - \text{cand}[k]\|) - \min_k (\|\mathbf{x}_T^{i+1}[j] - \text{cand}[k]\|)} \right);$ 
  end
   $j \leftarrow \arg \min(\text{costs});$ 
   $c \leftarrow \text{cand}[j];$ 
   $\mathbf{x}_I \leftarrow [\mathbf{x}_I, c];$ 
while  $\|\mathbf{x}_T^{i+1}[j] - c\| > \delta;$ 

```

---

[1] and another, with the same objective, but using electrocardiography signals as commands [17]; using voice commands to control a prosthetic robot arm [7]; controlling industrial machines through command voices [8]. Another work in robotic navigation tries to

include and process uncertain information in the voice commands [20], making it easier to learn and interact which gives to the user a better and natural experience. One drawback in these types of systems is that they tend to only focus on one perceptual modality, that is typically the sound.

## 5.2 Multimodal Representation Learning

Variational autoencoder-based (VAE) models are widely employed for the generative modelling of single-modality data in an unsupervised learning setting [10]. Several extensions of the original model by Kingma *et al.* have been proposed that consider the generative modeling of multimodal data. One class of approaches considers the *approximation* of single-modality latent distributions, enforcing the similarity between them with a statistical distance loss term [25, 30]. In a two-modality scenario, the Associative VAE (AVAE) model employs the symmetrical KL-divergence to learn a multimodal representation from two modality-specific latent distributions, suitable for cross-modality generation [30]. However, such approximation representation models struggle to scale to scenarios with more than two modalities as they require the instantiation of encoder networks for every possible combination of input modalities [11].

To allow learning a representation with higher number of modalities, another class of approaches considers learning a multimodal representation, by merging modality-specific information. The Multimodal VAE (MVAE) model employs a product-of-experts (PoE) solution to merge modality-specific information [29]. While scaling to large number of modalities, the model struggles to perform cross-modality generation from low-dimensional input information [23]. To address such robustness issue, the Mixture-of-Experts MVAE (MMVAE) model instead employs a mixture-of-experts (MoE) solution to merge information from multiple modalities [23]. However, the training of this solution incurs on a large computation cost, as it requires an importance weighted sampling training scheme. Recently, the Multimodal Unsupervised Sensing (MUSE) model was presented in the context of state representation for reinforcement learning agents in multimodal scenarios, allowing learning a multimodal representations scalable to a higher number of modalities and robust to missing modality information, by employing a PoE solution with a modified training scheme.

## 5.3 Dynamic Motion Control

There is an extend work in dynamic motion control that concerns about generating precise trajectories and control their dynamic parameters. DMPs, Gaussian Mixture Models and splines are examples of such methods. DMPs are used to encode a particular trajectory using stable attractor system (damped spring model), which make them robust to perturbations. Another related problem occurs when we want to create complex trajectory that can be decomposed into several smaller trajectories. In this case it is necessary to take into account the conversion of each individual trajectory but also how to handle the transition/joining from one trajectory to the next one, this problem is called DMP joining. A previous work [13] used a new DMP formulation where the goal function was replaced by a piecewise-linear function and the canonical system is a sigmoidal decay function and an updated nonlinear force-term, taking into account the new canonical system. This

**Table 4: Representation stage: generation of trajectory samples from all generative models. For  $R_{CVAE}(x_s, x_t)$  and  $R_{AVAE}(x_s, x_t)$ , the label information of a letter is given as input. For  $R_{AVAE}(x_i, x_t)$  the image of a letter is given as input. For  $R_{MUSE}(x_s, x_i, x_t)$  we generate samples using the two options: giving the label information and, also, the image of a letter.**

(a) $R_{CVAE}(x_s, x_t)$				(b) $R_{AVAE}(x_s, x_t)$				(c) $R_{MUSE}(x_s, x_i, x_t)$			
Input	"P"	"d"	"e"	Input	"P"	"d"	"e"	Input	"P"	"d"	"e"
Trajectory				Trajectory				Trajectory			

(d) $R_{AVAE}(x_i, x_t)$				(e) $R_{MUSE}(x_s, x_i, x_t)$			
Input				Input			
Trajectory				Trajectory			

**Table 5: Trajectory samples retrieved from running our full pipeline when given as input the sound of the respective word,  $x_s$ , or the images of the letters,  $x_i$ . For the Representation stage we tested all implemented generative models:  $R_{CVAE}(x_s, x_t)$ ,  $R_{AVAE}(x_s, x_t)$ ,  $R_{AVAE}(x_i, x_t)$ , and  $R_{MUSE}(x_s, x_i, x_t)$ .**

Model	input	word		
		bell	cat	jump
$R_{CVAE}(x_s, x_t)$	$x_s$			
$R_{AVAE}(x_s, x_t)$	$x_s$			
$R_{AVAE}(x_i, x_t)$	$x_i$			
$R_{MUSE}(x_s, x_i, x_t)$	$x_s$			
$R_{MUSE}(x_s, x_i, x_t)$	$x_i$			

work also uses a single set of overlapping kernels for the whole joint trajectory leading to smooth transitions between each individual trajectory. Finally, human handwriting trajectories were used to test the new approach.

## 6 CONCLUSION

In this work, we addressed the problem of *translating* multimodal commands, provided by different communication channels of the human user, to a sequence of movements executed by a robotic agent. We contributed with a novel three-stage pipeline that allows the processing, mapping, and generation of trajectory information, regardless of the communication channels employed by the human user to provide information to the agent. At the core of our

pipeline, we leverage multimodal generative models to learn a low-dimensional representation of the high-dimensional data provided by the human user, robust to partial observations. We instantiate our pipeline in the context of a Robotic Dictaphone: the generation of robotic handwriting from textual information provided through the speech of the human user. Our results show that our approach allows the generation of accurate handwritten samples, regardless of number and nature of the communication channels employed by the human user.

Our pipeline is agnostic both to the nature of the task and of the communication channels employed by the human user. In future work, we will extend our approach to scenarios with a higher number of modalities, addressing the scalability of our pipeline. Moreover, we will consider other tasks in human-robot collaboration scenarios, evaluating the role of multimodal command mapping for the effective execution of such tasks.

## REFERENCES

- [1] Hessam Ahmadi, Mohammad Saleh Hoseinzadeh, Ali Ekhlasi, and Mohamadreza Latifi. 2021. Voice commands classification in order to control robot movement. <https://doi.org/10.6084/m9.figshare.13712842.v2>
- [2] OpenAI: Marcin Andrychowicz, Bowen Baker, Maciek Chociej, Rafal Jozefowicz, Bob McGrew, Jakob Pachocki, Arthur Petron, Matthias Plappert, Glenn Powell, Alex Ray, et al. 2020. Learning dexterous in-hand manipulation. *The International Journal of Robotics Research* 39, 1 (2020), 3–20.
- [3] Claudine Badue, R nik Guidolini, Raphael Vivacqua Carneiro, Pedro Azevedo, Vinicius B Cardoso, Avelino Forechi, Luan Jesus, Rodrigo Berriel, Thiago M Paixao, Filipe Mutz, et al. 2021. Self-driving cars: A survey. *Expert Systems with Applications* 165 (2021), 113816.
- [4] Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations. *Advances in Neural Information Processing Systems* 33 (2020), 12449–12460.
- [5] Jessica Burgner-Kahrs, D Caleb Rucker, and Howie Choset. 2015. Continuum robots for medical applications: A survey. *IEEE Transactions on Robotics* 31, 6 (2015), 1261–1280.
- [6] Shruti Chandra, Raul Paradeda, Hang Yin, Pierre Dillenbourg, Rui Prada, and Ana Paiva. 2018. Do children perceive whether a robotic peer is learning or not?. In *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*. 41–49.
- [7] Koksul Gundogdu, Sumeyye Bayrakdar, and Ibrahim Yucedag. 2018. Developing and modeling of voice control system for prosthetic robot arm in medical systems. *Journal of King Saud University - Computer and Information Sciences* 30, 2 (2018), 198–205. <https://doi.org/10.1016/j.jksuci.2017.04.005>
- [8] Ruijia Huang and Guanglin Shi. 2012. Design of the control system for hybrid driving two-arm robot based on voice recognition. In *IEEE 10th International Conference on Industrial Informatics*. 602–605. <https://doi.org/10.1109/INDIN.>

- 2012.6300736
- [9] Auke Jan Ijspeert, Jun Nakanishi, Heiko Hoffmann, Peter Pastor, and Stefan Schaal. 2013. Dynamical Movement Primitives: Learning Attractor Models for Motor Behaviors. *Neural Computation* 25, 2 (02 2013), 328–373. [https://doi.org/10.1162/NECO\\_a\\_00393](https://doi.org/10.1162/NECO_a_00393) arXiv:[https://direct.mit.edu/neco/article-pdf/25/2/328/879555/neco\\_a\\_00393.pdf](https://direct.mit.edu/neco/article-pdf/25/2/328/879555/neco_a_00393.pdf)
  - [10] Diederik P Kingma and Max Welling. 2014. Auto-Encoding Variational Bayes. arXiv:1312.6114 [stat.ML]
  - [11] Timo Korthals, Daniel Rudolph, Jürgen Leitner, Marc Hesse, and Ulrich Rückert. 2019. Multi-Modal Generative Models for Learning Epistemic Active Sensing. In *2019 International Conference on Robotics and Automation (ICRA)*. 3319–3325. <https://doi.org/10.1109/ICRA.2019.8794458>
  - [12] Oliver Kroemer, Scott Niekum, and George Konidaris. 2021. A Review of Robot Learning for Manipulation: Challenges, Representations, and Algorithms. *J. Mach. Learn. Res.* 22 (2021), 30–1.
  - [13] Tomas Kulvicius, KeJun Ning, Miniya Tamosiunaite, and Florentin Worgötter. 2012. Joining Movement Sequences: Modified Dynamic Movement Primitives for Robotics Applications Exemplified on Handwriting. *IEEE Transactions on Robotics* 28, 1 (2012), 145–157.
  - [14] John E Laird, Kevin Gluck, John Anderson, Kenneth D Forbus, Odest Chadwicke Jenkins, Christian Lebiere, Dario Salvucci, Matthias Scheutz, Andrea Thomaz, Greg Trafton, et al. 2017. Interactive task learning. *IEEE Intelligent Systems* 32, 4 (2017), 6–21.
  - [15] Iolanda Leite, Carlos Martinho, and Ana Paiva. 2013. Social robots for long-term interaction: a survey. *International Journal of Social Robotics* 5, 2 (2013), 291–308.
  - [16] D. Llorens, F. Prat, A. Marzal, J. M. Vilar, M. J. Castro, J. C. Amengual, S. Barachina, A. Castellanos, S. España, J. A. Gómez, J. Gorbe, A. Gordo, V. Palazón, G. Peris, R. Ramos-Garijo, and F. Zamora. 2008. The UJIPenchars Database: a Pen-Based Database of Isolated Handwritten Characters. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)* (28-30), Nicoletta Calzolari (Conference Chair), Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, and Daniel Tapias (Eds.). European Language Resources Association (ELRA), Marrakech, Morocco, 2647–2651. <http://www.lrec-conf.org/proceedings/lrec2008/>.
  - [17] SA Lobov, VI Mironov, IA Kastalskiy, and VB Kazantsev. 2015. Combined Use of Command-Proportional Control of External Robotic Devices Based on Electromyography Signals. *Medical Technologies in Medicine/Sovremennyye Tehnologii v Medicine* 7, 4 (2015).
  - [18] Yuncheng Lu, Zhucun Xue, Gui-Song Xia, and Liangpei Zhang. 2018. A survey on vision-based UAV navigation. *Geo-spatial information science* 21, 1 (2018), 21–32.
  - [19] Francisco S Melo, Alberto Sardinha, David Belo, Marta Couto, Miguel Faria, Anabela Farias, Hugo Gamboa, Cátia Jesus, Mithun Kinarullathil, Pedro Lima, et al. 2019. Project INSIDE: towards autonomous semi-unstructured human-robot social interaction in autism therapy. *Artificial intelligence in medicine* 96 (2019), 198–216.
  - [20] M. A. Viraj J. Muthugala and A. G. Buddhika P. Jayasekara. 2016. Enhancing human-robot interaction by interpreting uncertain information in navigational commands based on experience and environment. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*. 2915–2921. <https://doi.org/10.1109/ICRA.2016.7487456>
  - [21] Javier Ruiz-del Solar, Patricio Loncomilla, and Naiomi Soto. 2018. A survey on deep learning methods for robot vision. *arXiv preprint arXiv:1803.10862* (2018).
  - [22] Brian Scassellati, Henny Admoni, and Maja Mataric. 2012. Robots for use in autism research. *Annual review of biomedical engineering* 14 (2012), 275–294.
  - [23] Yuge Shi, Siddharth N, Brooks Paige, and Philip Torr. 2019. Variational Mixture-of-Experts Autoencoders for Multi-Modal Deep Generative Models. In *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (Eds.), Vol. 32. Curran Associates, Inc., 15692–15703. <https://proceedings.neurips.cc/paper/2019/file/0ae775a8cb3b499ad1fca944e6f5c836-Paper.pdf>
  - [24] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. 2015. Learning structured output representation using deep conditional generative models. *Advances in neural information processing systems* 28 (2015), 3483–3491.
  - [25] Masahiro Suzuki, Kotaro Nakayama, and Yutaka Matsuo. 2016. Joint multimodal learning with deep generative models. *arXiv preprint arXiv:1611.01891* (2016).
  - [26] Stefanie Tellex, Thomas Kollar, Steven Dickerson, Matthew R Walter, Ashis Gopal Banerjee, Seth Teller, and Nicholas Roy. 2011. Approaching the symbol grounding problem with probabilistic graphical models. *AI magazine* 32, 4 (2011), 64–76.
  - [27] Miguel Vasco, Hang Yin, Francisco S. Melo, and Ana Paiva. 2021. How to Sense the World: Leveraging Hierarchy in Multimodal Perception for Robust Reinforcement Learning Agents. arXiv:2110.03608 [cs.LG]
  - [28] Valeria Villani, Fabio Pini, Francesco Leali, and Cristian Secchi. 2018. Survey on human-robot collaboration in industrial settings: Safety, intuitive interfaces and applications. *Mechatronics* 55 (2018), 248–266.
  - [29] Mike Wu and Noah Goodman. 2018. Multimodal generative models for scalable weakly-supervised learning. In *Advances in Neural Information Processing Systems*. 5575–5585.
  - [30] Hang Yin, Francisco Melo, Aude Billard, and Ana Paiva. 2017. Associate Latent Encodings in Learning from Demonstrations. *Proceedings of the AAAI Conference on Artificial Intelligence* 31, 1 (Feb. 2017), 3848–3854.
  - [31] Ekim Yurtsever, Jacob Lambert, Alexander Carballo, and Kazuya Takeda. 2020. A survey of autonomous driving: Common practices and emerging technologies. *IEEE access* 8 (2020), 58443–58469.