

Deep Movement Primitives: toward Breast Cancer Examination Robot

Oluwatoyin Sanni,^{1,*} Giorgio Bonvicini,² Muhammad Arshad Khan,¹ Pablo C. López-Custodio,¹
Kiyanoush Nazari,¹ Amir M. Ghalamzan E.^{1,*}

¹ University of Lincoln, UK, ² Polytechnic University of Milan, Italy
aghalamzanesfahani@lincon.ac.uk

Abstract

Breast cancer is the most common type of cancer worldwide. A robotic system performing autonomous breast palpation can make a significant impact on the related health sector worldwide. However, robot programming for breast palpating with different geometries is very complex and unsolved. Robot learning from demonstrations (LfD) reduces the programming time and cost. However, the available LfD are lacking the modelling of the manipulation path/trajectory as an explicit function of the visual sensory information. This paper presents a novel approach to manipulation path/trajectory planning called deep Movement Primitives that successfully generates the movements of a manipulator to reach a breast phantom and perform the palpation. We show the effectiveness of our approach by a series of real-robot experiments of reaching and palpating a breast phantom. The experimental results indicate our approach outperforms state-of-the-art method.

Breast cancer is the most common cancer worldwide with a significant impact on the life of patients and society (Ferry et al. 2021). 2.3 million females were diagnosed with breast cancer in 2020 with 685,000 related deaths worldwide (Sung et al. 2021). Less invasive and cheaper cancer treatments and higher quality of patients' lives post-cancer detection are among the benefits of early breast cancer detection (Kösters and Göttsche 2003). Breast Palpation (BP) is the easiest, most effective and most widely used early cancer detection method. BP – both self and clinical examinations – seeks to detect palpable anomalies in the breast tissue (Provencher et al. 2016; Saslow et al. 2004). BP involves human tactile and visual inspection during palpating the breast and lymph nodes. Because of the lack of patients' expertise in palpation, self-examination is ineffective across societies. Moreover, subjects are reluctant to be examined by human experts (Yang et al. 2010), detection precision depends on the examiner's expertise, and the availability of experts everywhere may be limited. Therefore, our survey shows autonomous robotic BP (ARB) is of great interest to both clinicians and patients (Houghton et al. 2021)

ARB reduces the burden on the general healthcare system for clinical BP, improves the accessibility of the ser-

vice and precision of early breast cancer detection. Although there are many studies for robotic tissue palpation for tumour detection (Scimeca et al. 2020; Herzig et al. 2018; Kobayashi et al. 2009; Nichols and Okamura 2015; Keshavarz and Mojra 2015), most of them focus on tissue stiffness classification and developing a suitable robot finger/hand for palpation. Hence, the problem of efficient path/motion planning for autonomous robotic palpation remains vastly unexplored. Geometrical variability across different palpation paths and the subjects' breasts leads to the complexity of motion/path planning for breast palpation.

Human experts suggest benchmark patterns for a more effective way of performing breast palpation (see Fig. 1a), such as circular, wedges or linear (Murali and Crabtree 1992). An autonomous robotic palpation system needs to encode the expert knowledge into the palpation movements and adapt the planned motions according to the breast geometry. Movement Primitives as a tool for compact representation of robot's control policy are used for generating robot motions (Ude et al. 2010; Matsubara, Hyon, and Morimoto 2011; Ghalamzan E. and Ragaglia 2018). Probabilistic Movement Primitives (ProMP) (Paraschos et al. 2013) can be used for planning and control purposes where it can express a distribution of trajectories by the corresponding variance and mean. Nonetheless, these conventional Movement Primitives methods (Paraschos et al. 2018, 2013; Schaal 2006; Ude et al. 2010) cannot capture the correlation between the visual information and the generated movements.

We propose a novel Learning from Demonstration (LfD) approach called *deep movement primitives* (**deep-MP**)¹ directly mapping the visual sensory information into the learned trajectory. Other contributions of this paper include: (i) the effectiveness of deep-MP variations for different tasks complexities are extensively studied; (ii) a series of real-robot breast palpation experiments that show the effectiveness of our proposed approach to complex trajectory/path planning tasks.

Related Works

In trajectory planning, LfD methods use demonstrations to build a task model and execute it. Probabilistic approaches

*Authors contributed equally.
Copyright © 2022, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

¹Code, data and an extended appendix are available here:
https://github.com/imanlab/deep_movement_primitives

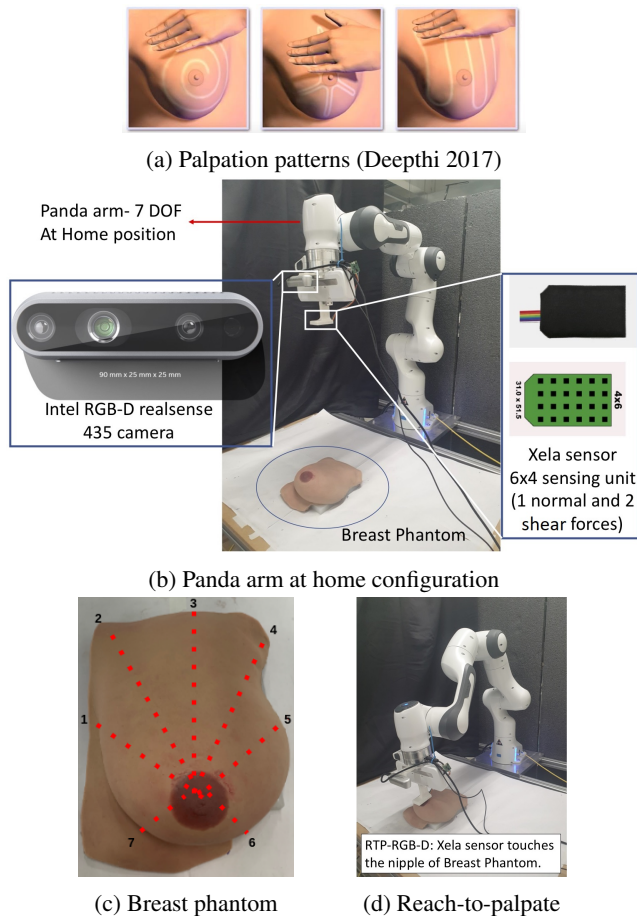


Figure 1: Breast palpation is an efficient and the first approach to evaluate glandular tissue and nipple (Mahoney and Csima 1982): (a) well-known breast palpation paths: Spiral (left), Wedges (middle) and Lines (right); (b) 7-DOF Panda arm at Home Configuration; (c) the breast phantom used for study in this paper and the demonstrated palpation paths. The phantom is made of silicon with 2 lumps implanted on path 4 and 7; (d) reach-to-palpatate (RTP) action of Panda arm; RGB-D images taken at home configuration shown in Fig. 1b.

such as Gaussian Mixture models (GMM) (Jaquier, Ginsbourger, and Calinon 2019; Girgin et al. 2020), Gaussian Process (Schneider and Ertel 2010) are applied in LfD settings for expressing the distributions of trajectories by the corresponding mean and covariance. However, these models do not encode the relation between the visual information of robot’s workspace and the demonstrated behaviour (Rana et al. 2018).

End-to-end learning-based control approaches such as Inverse Reinforcement Learning (Levine, Popovic, and Koltun 2011), vision-based Model Predictive Control (Finn and Levine 2017), or Behaviour cloning (Rahmatizadeh et al. 2018) usually suffer from lack of generalisation and they are only applied to a class of tasks involved complicated motion control but not complex motion planning. End-to-end LfD enables generating control commands directly from

raw image data, e.g., (Rahmatizadeh et al. 2018) proposed behaviour cloning probabilistic generative model. However, high dimensional observation space, e.g. raw images, make the combined motion/path planning and motion control intractable in such settings (Akgun et al. 2012; Nagahama and Yamazaki 2019). In a line of research, deep-time-series are used to learn the control policy of a robot to perform a specific task. For instance, (Levine et al. 2018, 2016) applied Deep Neural Networks (NNs) combined with Recurrent NNs to image data to learn robots’ control policy. In these frameworks, deep NNs finds a mapping between the raw image and desired control signals. These approaches are proved to be effective only for limited classes of tasks with challenging motion control. Moreover, such models are hard to interpret.

Dynamic Movement Primitives (DMP) model (Schaal 2006) is a well-known LfD approach useful for imitating a single demonstrated trajectory. Recent works on deep DMP (d-DMP) show a Deep NN learning to generate the parameters of a DMP model from an image (Ridge et al. 2020; Pervez, Mao, and Lee 2017). For instance, (Ridge et al. 2020) propose a neural network that is trained to output the parameters of DMP (Schaal 2006). (Pervez, Mao, and Lee 2017) also use deep NNs to learn the forcing terms of the DMP model for visual servoing. A distribution of a set of demonstrations expresses variability of the task executions which is captured by Probabilistic Movement Primitives (ProMP) (Paraschos et al. 2018, 2013). However, it fails to capture the relation between visual information and trajectory variations, hence, it misses the generalisation capability (Rueckert et al. 2015).

Breast palpation movements is hard to program and is needed for ARBP (Scimeca et al. 2020; Herzig et al. 2018). Our proposed method can address the existing challenges in motion planning for Breast Palpation, such as (1) variations across different palpation executions; (2) variability across different palpation paths; (3) relation between the visual information and demonstrated trajectories.

In contrast to d-DMP (Pervez, Mao, and Lee 2017; Ridge et al. 2020) being deterministic – i.e. they cannot capture the distribution of trajectories and are designed only to generalise to initial and goal points – and ProMP, which only generalise to initial, goal and via points, our approach successfully maps images of breast phantom in robot workspace to robot joint space trajectories. Deep movement primitives utilises deep Neural Networks (NNs) for feature extraction and subsequent generation of the weights defining a ProMP trajectory. Our customised loss function is defined as the distance between the Ground Truth (GT) and the trajectory generated by the predicted weights, rather than being the distance between GT and the predicted weights themselves. We implemented several different network and LfD architectures. The results obtained by our framework show its effectiveness for breast palpation tasks in a practical and real-world use case. We consider two sub-tasks for breast palpation. *Reach to Palpatate (RTP)*: RGB-D images of the breast phantom are used to generate trajectory weights that control the robot joints to reach the starting point of palpation on the breast phantom. *Wedges Palpation Path (WPP)*:

deep-MP is used to generate WPP shown in Fig. 1c) from a dataset of demonstrations. This framework will be integrated into a motion control in future works.

Problem Formulation

Consider a set of N_{tr} demonstrations, which is defined as $\mathcal{T} := \{\{\mathbf{Q}^1, \mathbf{I}^1\}, \dots, \{\mathbf{Q}^{N_{\text{tr}}}, \mathbf{I}^{N_{\text{tr}}}\}\}$ where \mathbf{Q}^n , $n = 1, \dots, N_{\text{tr}}$, are the joint space trajectories and \mathbf{I}^n is RGB-D images taken from the corresponding robot's workspace. For a single joint, we define a trajectory as the ordered set $\mathbf{q} := \{q_t\}_{t=1, \dots, T}$, where $q_t \in \mathbb{R}$ is the joint position at sample t , and $\mathbf{Q} := \{\mathbf{q}_1, \dots, \mathbf{q}_{N_{\text{joint}}}\}$ where N_{joint} is the number of the joints of the manipulator.

Probabilistic Movement Primitives (ProMP) In order to define a distribution over trajectories, We first model a trajectory with an observation uncertainty added to the following deterministic model (Paraschos et al. 2013):

$$q = \sum_{i=1}^{N_{\text{bas}}} \theta_i \psi_i(z(t)) + \epsilon_q \quad (1)$$

where ψ_i are basis functions (usually Gaussian (Bishop 2006)) evaluated at $z(t)$. z is a phase function that allows time modulation. If no modulation is required, then $z(t) = t/f$, where f is the sampling frequency. $\theta_i \in \mathbb{R}$ are weights, and ϵ_q adds zero-mean Gaussian observation noise with variance Σ_q .

For stroke-like movements, the following normalised Gaussian basis functions are used:

$$\psi_i(t) := \frac{b_i(z(t))}{\sum_{j=1}^{N_{\text{bas}}} b_j(z(t))} \quad (2)$$

where

$$b_i(z(t)) := \exp\left(-\frac{(z(t) - c_i)^2}{2h}\right) \quad (3)$$

We can also write eq. (1) in a matrix form, as follows:

$$q_t = \Psi_t^T \Theta + \epsilon_q \quad (4)$$

where $\Psi_t := (\psi_1(z(t)), \dots, \psi_{N_{\text{bas}}}(z(t))) \in \mathbb{R}^{N_{\text{bas}} \times 1}$, $\Theta := (\theta_1, \dots, \theta_{N_{\text{bas}}}) \in \mathbb{R}^{N_{\text{bas}} \times 1}$, and we also define $\Omega := (\Omega_1, \dots, \Omega_{N_{\text{joint}}}) \in \mathbb{R}^{N_{\text{bas}} \times N_{\text{joint}} \times 1}$ and $\Phi := [\Psi_1, \dots, \Psi_T]^T \in \mathbb{R}^{T \times N_{\text{bas}}}$.

From eq. (1), it follows that the probability of observing q_t is given by:

$$p(q_t | \Theta) = \mathcal{N}(q_t | \Psi_t^T \Theta, \Sigma_q) \quad (5)$$

Since Σ_q is the same for every time step, the values q_t are taken from independent and identical distributions, i.i.d. Hence, the probability of observing a trajectory \mathbf{q} is given by:

$$p(\mathbf{q} | \Theta) := \prod_{t=1}^T p(q_t | \Theta) \quad (6)$$

However, since parameters Θ are to be learnt from data, we also assume such parameters are taken from a distribution $\Theta \sim p(\Theta | \rho) = \mathcal{N}(\Theta | \mu_\Theta, \Sigma_\Theta)$. We therefore would

like to have a predictive distribution of q_t which does not depend on Θ , but on $\rho := (\mu_\Theta, \Sigma_\Theta)$. This is done by marginalising Θ out in the distribution as follows:

$$\begin{aligned} p(q_t | \rho) &= \int \mathcal{N}(q_t | \Psi_t^T \Theta, \Sigma_q) \mathcal{N}(\Theta | \mu_\Theta, \Sigma_\Theta) d\Theta \\ &= \mathcal{N}(q_t | \Psi_t^T \mu_\Theta, \Sigma_q + \Psi_t^T \Sigma_\Theta \Psi_t) \end{aligned} \quad (7)$$

deep-MP weights learning: The weights of ProMP models are conventionally learned from demonstrations where they can be later adapted according to different trajectory reproduction needs, e.g. (1) the initial/goal point of the desired trajectory are set, or (2) some via points are determined based on the problem constraints. Computing the via/start/goal points based on visual sensory information of robot's workspace needs hand-designed and task- or robot's workspace-specific features. This is effective and handy in some robotic tasks like simple pick-and-place, but it is too complex for breast palpation, e.g., in which the task trajectory and the geometry of the breast are related. Two following deep-MP models learn the relation between visual sensory information and joint trajectories.

(deep-MP): Instead of learning the weights of the ProMP, a deep model— that can be a CNN, FC or PointNet model— captures the correlation between the visual sensory information and ProMP weights as per eq. (8). The algorithm is described in Alg. 1. Moreover, a schematic of the algorithm is shown in Fig. 2 in the green block at top right).

For a neural network similar to Fig. 2, network's behaviour can be described by the following:

$$\Theta_k = h_k(\mathbf{W}_k, \mathbf{I}_k, \sigma_k) + v_k \quad (8)$$

Equation (8), known as observation equation, shows that the network's target vector Θ_k is equivalent to a nonlinear

Algorithm 1: Deep MP

Input: NN architecture h , ProMP basis functions Φ , image \mathbf{I} , training set trajectories \mathbf{q} , activation function σ
Output: NN weights \mathbf{W} , predicted trajectory $\hat{\mathbf{q}}$
Note: This pseudo code is for single joint trajectory $\hat{\mathbf{q}}$. Generalising it for all joints $\hat{\mathbf{Q}}$ is straightforward.

- 1: *Dataset:* $\mathcal{T} \leftarrow \{\mathbf{q}, \mathbf{I}\}_{1, \dots, N_{\text{tr}}}$
 - 2: *InitialiseDeepModel:* $\hat{\Theta} \leftarrow h(\mathbf{W}, \mathbf{I}, \sigma)$ eq. (8) as per Fig. 4: either CNN (2-D) or FCN (1-D);
 - 3: *InitialiseProMP:* $\hat{\mathbf{q}} \leftarrow \Phi^T \hat{\Theta}$ (eq. (4))
 - 4: *RMSE* $\leftarrow e = \|\hat{\mathbf{q}} - \mathbf{q}\|$
 - 5: **while** ($e > \epsilon$) **do**
 - 6: **for all** $\{\mathbf{q}, \mathbf{I}\} \in \mathcal{T}$ **do**
 - 7: FORWARDPROPAGATION: $\hat{\Theta}_k = h(\mathbf{W}_k, \mathbf{I}_k, \sigma)$
 - 8: FORWARDPROMP: $\hat{\mathbf{q}}_k = \Phi^T \hat{\Theta}_k$ (eq. (4))
 - 9: RMSEJOINTLOSS: $e_k = e_{k-1} + \|\hat{\mathbf{q}}_k - \mathbf{q}_k\|$
 - 10: **end for**
 - 11: BACKPROPAGATION: $\mathbf{W}_{k+1} \leftarrow \{\mathbf{W}_k, \frac{\partial e_k}{\partial \mathbf{W}_k}\}$
 - 12: **end while**
 - 13: deep-MP: $\hat{\mathbf{q}} = h(\Phi, \mathbf{W}, \mathbf{I}, \sigma)$
 - 14: **end**
-

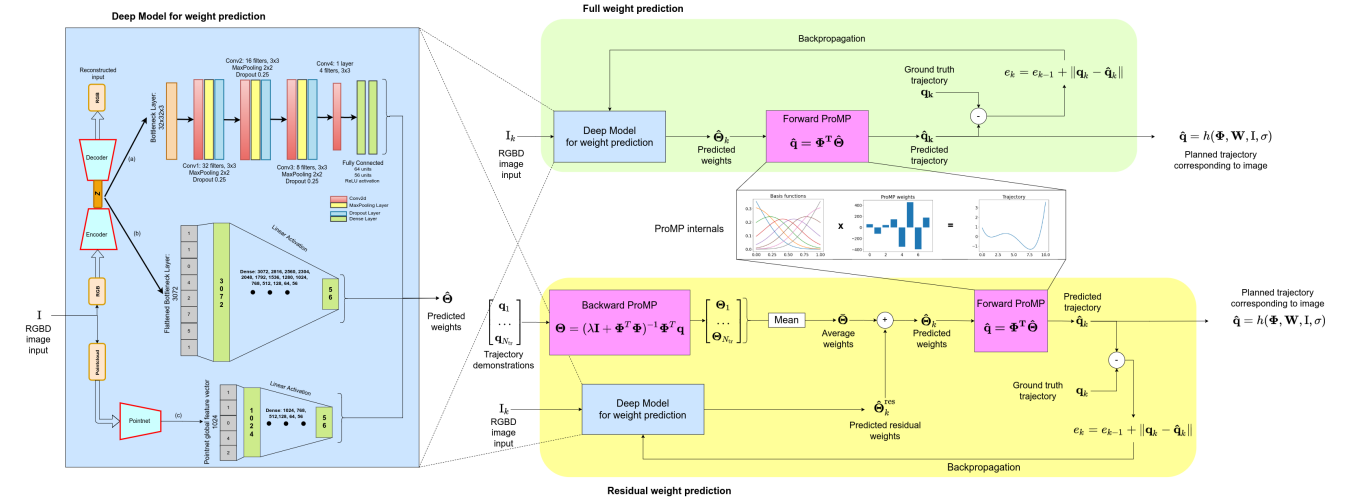


Figure 2: The blue box contains three NN models learning the weight of ProMP. RGB data is passed through an autoencoder, producing a bottleneck representation of the input Z , which is then fed into a CNN (a) or flattened and fed into a Fully Connected network (b). Alternatively, the depth data from the image, in the form of vector of (x, y, z) coordinates of the pointcloud, is fed to a PointNet network (c) and the resulting feature vector is passed through a dense network. All models are used to predict the full ProMP weights (green box - Alg. 1) or just the residual with respect to the mean (yellow box - Alg. 2). Fig. A.4 in the Appendix is an higher resolution version of this image.

function h_k of the input image I_k , the weight parameter \mathbf{W}_k , the node activation σ_k and the observation/measurement noise v_k . We consider the measurement noise to be a zero-mean white noise with covariance given by $E[v_k v_k^T] = R_k$.

Algorithm 2: Residual deep MP

Input: NN architecture h , ProMP basis functions Φ , image I , training set trajectories \mathbf{q} , activation function σ
Output: NN weights \mathbf{W} , predicted trajectory $\hat{\mathbf{q}}$
Note: This pseudo code is for single joint trajectory $\hat{\mathbf{q}}$. Generalising it for all joints $\hat{\mathbf{Q}}$ is straightforward.

```

1: Dataset:  $\mathcal{T} \leftarrow \{\mathbf{q}, I\}_{1, \dots, N_{tr}}$ 
2: InitWeights:  $\{\Theta_1, \dots, \Theta_{N_{tr}}\} \leftarrow \Phi \cdot \{\mathbf{q}_1, \dots, \mathbf{q}_{N_{tr}}\}$ 
3: InitAverages:  $\bar{\Theta} \leftarrow \text{mean}(\Theta_1, \dots, \Theta_{N_{tr}})$  (eq. (9))
4: InitDeepModel:  $\hat{\Theta}^{\text{res}} \leftarrow h(\mathbf{W}, I, \sigma)$  eq. (8)
   as per Fig. 4: either CNN (2-D) or FCN (1-D);
5: InitFullWeights:  $\hat{\Theta} = \hat{\Theta}^{\text{res}} + \bar{\Theta}$  (eq. (10))
6: InitProMP:  $\hat{\mathbf{q}} \leftarrow \Phi \hat{\Theta}$  (eq. (4))
7: RMSE:  $e = \|\hat{\mathbf{q}} - \mathbf{q}\|$ 
8: while ( $e > \epsilon$ ) do
9:   for all  $\{\mathbf{q}, I\} \in \mathcal{T}$  do
10:    FORWARDPROPAGATION:  $\hat{\Theta}_k^{\text{res}} = h(\mathbf{W}_k, I_k, \sigma)$ 
11:    FULLWEIGHTS:  $\hat{\Theta} = \hat{\Theta}^{\text{res}} + \bar{\Theta}$  (eq. (10))
12:    FORWARDPROMP:  $\hat{\mathbf{q}}_k = \Phi \hat{\Theta}_k$  (eq. (4))
13:    RMSEJOINTLOSS:  $e_k = e_{k-1} + \|\hat{\mathbf{q}}_k - \mathbf{q}_k\|$ 
14:   end for
15:   BACKPROPAGATION:  $\mathbf{W}_{k+1} \leftarrow \{\mathbf{W}_k, \frac{\partial e_k}{\partial \mathbf{W}_k}\}$ 
16: end while
17: deep-MP residual:  $\hat{\mathbf{q}} = h(\Phi, \mathbf{W}, I, \sigma)$ 
18: end

```

where h_k is a nonlinear deep model mapping the image I_k taken by robot's camera at a home position (see Fig. 1b) to the ProMP weights. The weights then generate the corresponding trajectories using eq. (4).

(Residual deep-MP): usually a set of demonstrated trajectories convey information about the presented behaviour regardless of the scene. GMM, GP and ProMP have been used to encode such information in a probabilistic model that can be expressed as a mean and distribution (see Fig. 3). In order to improve the performance of the deep-MP, we propose a deep model which learns the correlation between the input image and residual trajectories, i.e. difference between the mean and demonstrated trajectories. Hence, the deep model is required to learn a much simpler mapping since part of the complexities are captured by the mean trajectory (see the yellow block at bottom right in Fig. 2). First, we learn to fit a ProMP trajectory $\hat{\mathbf{Q}}$ to the demonstrated trajectories using least squares optimisation.

Hence, we can compute the mean weights $\bar{\Theta}$ and mean joint space trajectories \mathbf{q} (see Fig. 3) by maximising the likelihood as per eq. (9).

$$\Theta^n = (\lambda \mathbf{I} + \Phi^T \Phi)^{-1} \Phi^T \mathbf{q}^n, \forall n = 1, \dots, N_{tr} \quad (9)$$

$$\bar{\Theta} = \mathbb{E}([\Theta^1, \dots, \Theta^{N_{tr}}])$$

where $\mathbf{q}^n := (q_1, \dots, q_T) \in \mathbb{R}^{T \times 1}$ is the vectorized form of single-joint values in trajectory n , and λ is a regularising term used to avoid over-fitting in the original optimisation objective. Then the deep-MP model learns the correlation between the residuals of the trajectories and the visual sensory information.

$$\Theta^n = \Theta^{\text{res}, n} + \bar{\Theta} \quad (10)$$

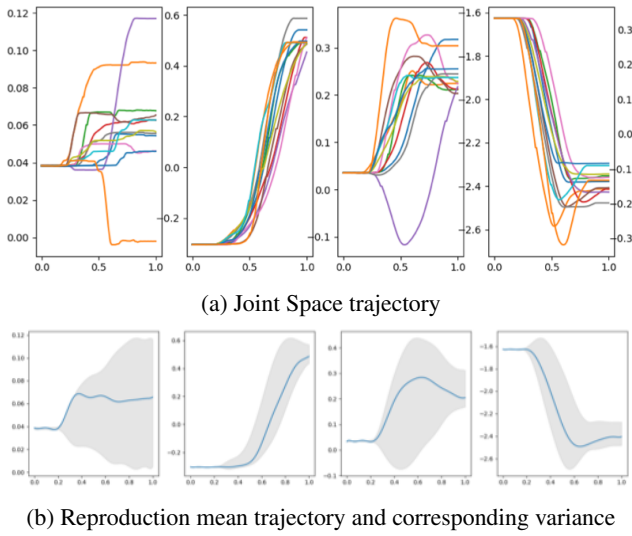


Figure 3: (a) Samples of RTP-RGBD demonstrated trajectories from left/joint 1 to right/joint 4 – trajectories of joint 5, 6 and 7 are not shown here; (b) the computed mean (solid blue lines)– used for the residual deep-MP implementation– and representative variations of the distribution (shaded grey areas).

where Θ^n and $\Theta^{\text{res},n}$ are, respectively, the full and residual weights of the ProMP model for N_{tr} demonstrated trajectories. We can use the deep-MP model presented in eq. (8) to learn $\Theta^{\text{res},n}$ which will be added to the Θ to form the ProMP corresponding with $\{\mathbf{Q}^n, \mathbf{I}^n\}$, as per eq. (4).

Hardware setup and data collection

Our experimental setup consists of a 7-DoF Panda robotic arm manufactured by Franka Emika. An Intel RealSense D435i RGB-D camera is mounted on the wrist of the arm. We also use a tactile finger consisting of a 6x4 uSkin Xela magnetic-based tactile sensor for RTP-RGBD and WPP data collection. This sensor is firmly connected to the left finger link of the gripper using a 3D printed mount. Although the reading of the tactile sensing is not used in this study, we will use it for future study of palpation motion control.

We have obtained three data sets: (1) we collected reach-to-palpatate (RTP) dataset, called RTP-RGB, in a mock study; (2) Reach-to-palpatate dataset 2 called RTP-RGBD, and (3) Wedged-palpation-path dataset called WPP.

Reach-to-palpatate (RTP) (i) RTP-RGB: Our mock study includes the RTP-RGB data collection. For each sample in this dataset, the robotic arm starts from a fixed home pose as shown in Fig. 1b. At this home pose, the camera takes an RGB image of the breast phantom; and then the robot is manually moved to the corner of the breast phantom in kinesthetic teaching mode as shown in Fig. 4. This dataset contains 500 samples, i.e. the robot at home configuration takes RGB images of breast phantom at a random position on the table. We have trained the CNN and FC deep-MP model, where they yield 0.0108 and 0.0118 [Radian²] errors

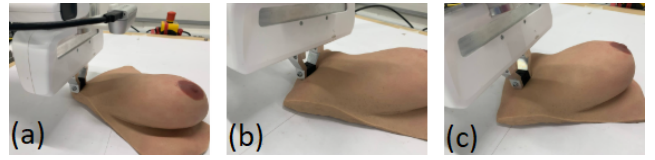


Figure 4: RTP-RGB Datasets: CNN–deep-MP Model tested on unseen Breast Phantom configurations. (a) shows precise task execution whereas (b) and (c) have a larger errors as they belong to regions with different sample densities (a, b and c belong to region 1, 2 and 4, respectively in Fig. A.3 of Appendix). The distance between the Robot EE and the corner of the breast phantom–the desired touching point demonstrated– is because of difference in sample density of the corresponding regions.

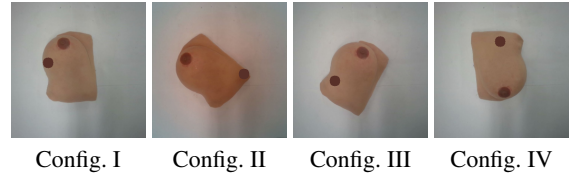


Figure 5: Samples of WPP data set: the brown disk on images show the desired end-points of palpation path 5, 1, 4 and 3 at configuration I, II, III and IV, respectively. The starting point for each demonstrated palpation is the nipple.

in joint space and 39.7 mm and 46.8 mm errors in task space.

Full details of the RTP-RGB dataset and the obtained results are described in Appendix. The results obtained by this dataset suggest (1) we need a more structured dataset to better understand the impact of samples density on the results; (2) the depth data may be relevant; (3) a more challenging task is needed to showcase the effectiveness of the approach.

(ii) RTP-RGBD: the setup for the following data collection is the same as the one in RTP-RGB dataset. Nonetheless, we collected RGB and depth data for each sample and the robot is moved by joint space motion planning to the nipple of the breast phantom. We consider 4 regions for data collection as shown in Fig. 6: Region A, B, C and D. After each sample collection, the breast phantom was moved to a new location within the region boundary to create uniform distribution for each region with different densities as shown in Fig. 6. A total of 545 samples were collected with 292, 128, 73 and 52 samples in region A, B, C, and D respectively.

Wedges Palpation Path (WPP) After moving the robot tactile finger to the nipple of the breast phantom, the robot needs to follow the palpation path. The robot is moved using kinesthetic-teaching-mode from nipple along to the edge of the phantom similar to WPP shown in Fig. 1c. Synchronised robot full state, tactile sensor readings, and joint trajectory are recorded. 31 palpation trials for every 7 WPPs (Fig. 1c) and four different phantom configurations (Fig. 5) are recorded. A total of 868 palpation samples were collected in WPP dataset.

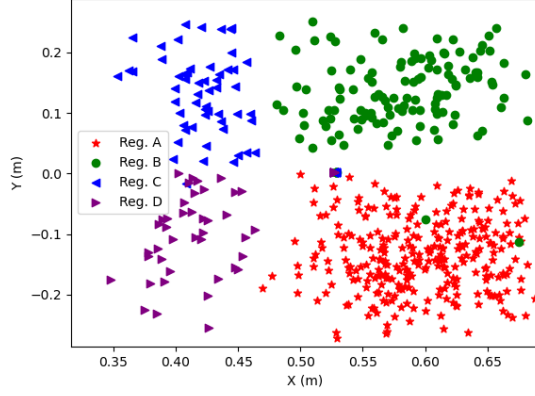


Figure 6: XY coordinates of the end-effector when the robot reaches the start point of the palpation in RTP-RGBD dataset.

Experiments and Results

To validate our hypothesis, i.e. deep-MP can produce accurate trajectory/path for breast palpation, we performed different experiments using the collected dataset RTP-RGBD and WPP in this proof-of-concept study. Although we only have one breast model and we cannot claim the generalisation across different breast geometry, the palpation paths varies across different palpation experiments in WPP experiments due to variations in the palpation terminal points.

Autoencoder as feature extractor: Using the RGB images, we trained an autoencoder on the 500 images for feature extraction, by using the bottleneck layer of the trained autoencoder. We then trained two models that use the learned features from the bottleneck layer to predict the joint weights that takes the robot to the palpation starting point (see Fig. 2).

Convolutional Neural Network (CNN) for mapping from features to ProMP weights Model: The bottleneck layer was an image of 32x32x3 which was passed through a CNN architecture to predict the joint weights. We implemented two versions of this model, one using Alg. 1 that predicts the full ProMP weights and one using Alg. 2 that predicts the residual ProMP weights with respect to the mean trajectory for any given region.

Fully Connected (FC) Model for mapping from features to ProMP weights: The bottleneck layer was flattened to a 1-D vector and then fed into stack of dense layers to predict the joint weights at the last layer.

PointNet for mapping the pointcloud coordinates to ProMP weights: In addition to the CNN and FC models described above, we also trained deep models using the pointcloud coordinates. The vector of Cartesian coordinates was fed into a pre-trained PointNet network (Qi et al. 2017), from which the global feature vector was extracted and fed into a smaller Fully Connected network to predict the joint weights (Blue block in Fig. 2).

For all models, the dataset is split into 85% and 15% for training and testing, respectively, and 25% of training data are used for cross-validation. We use Tensorflow framework

for training with Adam optimiser using a learning rate of 0.001 and a batch size of 32. Our models were trained using 150 epochs with a callback function to terminate training when over-fitting.

Loss function We have implemented our custom loss function using the deep-MP model. We denote the ground truth of our joints weights as $\Theta_{gt} \in \mathbb{R}^{N_{bas} \times 1}$ and the corresponding predicted ones as $\Theta_{ps} \in \mathbb{R}^{N_{bas} \times 1}$. Our loss function is the root mean squared error between the trajectories generated by Θ_{gt} and Θ_{ps} . The loss of a predicted vector Θ_{ps} is then given by:

$$L(\Theta_{ps}, \Theta_{gt}) := \sqrt{\left(\frac{1}{T}\right) \sum_{t=1}^T (q_{gt,t} - q_{ps,t})^2} \quad (11)$$

where $q_{gt,t} = \Phi_t^T \Theta_{gt}$ and $q_{ps,t} = \Phi_t^T \Theta_{ps}$.

For our experiments, the number of samples was $T = 150$.

Metrics To measure the performance of the approach, N_{test} test samples are evaluated. Let $\Theta_{ps}^n \in \mathbb{R}^{N_{bas} \times 1}$ be the predicted weights of test sample n . We consider the following two metrics.

(i) Average Mean Squared Error (AveMSE) between the trajectories generated from the predicted weights and ground truth.

$$\text{AveMSE} := \frac{1}{N_{test}} \sum_{n=1}^{N_{test}} [L(\Theta_{ps}^n, \Theta_{gt}^n)]^2 \quad (12)$$

where N_{test} is the number of samples in the test set.

(ii) Average Euclidean Distance (AveED) between the position of the end-effector at the last configuration, $t = T$ generated by deep-MP model and the ground truth.

$$\text{AveED} := \frac{1}{N_{test}} \sum_{n=1}^{N_{test}} |\mathbf{r}_{ee,ps}^n - \mathbf{r}_{ee,gt}^n| \quad (13)$$

where, $\mathbf{r}_{ee,gt}^n, \mathbf{r}_{ee,ps}^n \in \mathbb{R}^{3 \times 1}$ are, respectively, the ground truth and predicted position vectors of the end-effector for sample n , at $t = T$.

The CNN network architectures and the loss function for the WPP and RTP are the same, except the last units of the dense layer in CNN-RTP is replaced with 70 units in CNN-WPP. Moreover, the images input to autoencoder are marked with the target point as shown in Fig. 5. The training parameters for WPP and RTP experiments are the same. However, 200 epochs was used in WPP.

d-MDP (Ridge et al. 2020) recently proposed deep DMP. We also implemented d-MDP both for RTP and WPP tasks with our best CNN (see Appendix for implementation details). The results obtained by d-MDP and deep-MP are shown in Table 2 for WPP experiments illustrating our deep-MP models performance is much better than the d-MDP ones in all the cases.

While the CNN deep-MP outperforms the FC, d-MDP and PointNet deep-MP model in RTP-RGBD dataset in RTP

Model	Reg. A (292 samples)	MSE*			Absolute Error**			
		B (128)	C (73)	D (52)	A (292)	B (128)	C (73)	D (52)
CNN residual deep-MP	0.0009	0.0030	0.0051	0.0047	24.6	41.7	72.1	72.5
CNN deep-MP	0.0016	0.0031	0.0037	0.0041	41.1	48.0	55.2	58.5
FC deep-MP	0.0047	0.0051	0.0058	0.0082	47.2	63.3	69.5	83.8
PointNet deep-MP	0.0095	0.0169	0.0233	0.0191	75.8	192.3	226.8	167.7
d-DMP	0.0117	0.0109	0.0172	0.0226	84.8	66.5	113.8	132.6

* MSE in rad^2 ; ** Absolute error in mm;

Table 1: Evaluation of RTP models with Average MSE (AveMSE) of the joint trajectory in each region of RTP-RGBD data set (columns at left); Evaluation of RTP models with Average Euclidean Distance (AveED) in cartesian coordinates of the end effector position at the end of the trajectory in each region of RTP-RGBD data set (columns at right).

Model	MSE*				Absolute Error**			
	Config.I	Config.II	Config.III	Config.IV	Config.I	Config.II	Config.III	Config.IV
CNN deep-MP	0.0373	0.0245	0.0072	0.0420	67.4	57.6	53.9	63.1
CNN residual deep-MP	0.0380	0.0763	0.0662	0.0465	61.3	58.2	89.3	83.6
d-DMP	1.0541	1.4307	1.7214	1.7013	487.0	499.9	399.8	407.9
WPP4 CNN deep-MP	0.0340	0.0617	0.0441	0.0756	63.8	112.9	113.5	61.4

* MSE in rad^2 ; ** Absolute error in mm.

Table 2: Evaluation of CNN deep-MP and CNN residual deep-MP Model with AveMSE and Absolute Error at each configuration for WPP9 data set (middle row). Evaluation of CNN deep-MP for WPP4 dataset (bottom row).

experiments (Table 1), the performance of d-DMP is better than just some cases of PointNet deep-MP. CNN can capture more complex mapping than FC model and PointNet deep-MP; hence, it is by far better than others (the trajectories generated by the CNN models are shown in Fig. A.5 and A.6 of Appendix).

We considered different configurations of training and test set for WPP task conducting a series of experiments – {WPP1, . . . , WPP10} (see Appendix for details). Two experiments with interesting results are presented in Table 2. Our results (see Table A.4 in Appendix) show the best model performance is obtained when including all the palpation path, as it was done in WPP9 experiment. The results of WPP9 are also presented in the middle row of Table 2 showing all deep-MP models outperforming d-DMP by a large margin. Another interesting observation is presented in the bottom row of Table 2 for WPP4 where – similarly to WPP1-3 – only palpation 1,4,5 are seen in training and 2,3 used for testing. The results indicate that our model is able to generalise to unseen geometry of breast palpation. Other eight ablation studies also suggest the effectiveness of our deep-MP model for complex palpation tasks.

The results in Table 1 and 2 show that residual version of deep-MP outperforms the standard one in RTP but not quite well in WPP. We learned that RTP trajectories are more uniform than WPP ones. Moreover, the RTP has travelling distance longer than WPP with much more non-linearity. Hence, the mean trajectory across different RTP demonstrations is more informative than the WPP. i.e. the mean trajectory captures large non-linearity across the RTP trajectories, hence the CNN model requires to learn much simpler map-

ping.

In spite of WPP results (in Table 2), the MSE and Absolute Error of d-DMP and the variations of deep-MP reported in RTP cases (in Table 1) are very close. Hence, the combined results in Table 1 and Table 2 suggest that d-DMP can perform relatively well in simple tasks such as RTP – where, in contrast to WPP, RTP tasks depend only on a target point, i.e. the nipple – whereas it yields a very poor performance in complex tasks such as WPP. The video attachment shows deep-MP successfully outputting the joints weights for palpation paths from the nipple to a terminal point that is set manually – demonstrating the generalisation of our model – or autonomously.

Conclusion

Autonomous robots for breast palpation can have significant impact on the societies health sector. One of the challenges for such technologies is robot programming which forms a very interesting scientific problem. We proposed a novel learning from demonstration method, called deep movement primitives, that maps visual information to palpation trajectories and is useful for programming a robot for autonomous palpation. We show the effectiveness of our approach in a series of real-robot breast phantom palpation experiments. While state-of-the-art approach fails to learn palpation due to the variations in the start, terminal points and the intermediate points, our results shows deep-MP is a suitable method for learning complex tasks, such as palpation. Our future works include use of the sensor data and creating demonstrations across different breast models that helps deep-MP generalise even better.

Ethical statement

Cancer Research UK has funded a project in 2020 to shape new technology for breast health called ARTEMIS². This proof of concept project aims to develop a breast cancer examination robot in collaboration with clinicians and with Patient and Public Involvement (PPI) (Houghton et al. 2021). The consortium consists of University of Lincoln, University of Bristol and Imperial College London. Our partners are developing a soft robot tool and soft sensors for safe, non-invasive and comfortable use, building Breast phantoms based on real subjects' breasts that can give clinicians a feel of real breasts during palpation, collecting data of clinicians performing palpation, with active PPI in design and development of the technology.

The autonomous Breast Palpation Robot (ABPR) will have a chair where patients can incline/bend forward 45 degrees and lie on the top of the robot housing and breasts go into designated holes allowing the soft sensors to touch the breast and perform the palpation. A survey of 155 women in the United Kingdom was conducted, showing them schematics of different designs of ABPR. Results indicated enthusiasm for ABPR with 92% of respondents indicating they would use ABPR. 83% would be willing to be examined for up to 15 minutes. GP surgery is the most popular location for ABPR. Thematic analysis of free-text responses identified the following: a) Subjects perceived ABPR has the potential to address limitations in current screening services; b) ABPR facilitates increased user choice and autonomy; c) there are ethical motivations for supporting ABPR development; d) accuracy is essential; e) integration with health services is important. The above results are included in a paper (Houghton et al. 2021) under revision for final publication.

ABPR collects data for clinician references and potentially reduces the number of (true negative) visits to hospitals for breast cancer examination. It is no replacement for hospital examinations. We aim to develop a cheap device safe, comfortable, reliable and accessible—especially in poor communities/countries that lack hospitals or expert clinicians.

ABPR allows recording the history of the palpation data for individuals helping patients and clinicians with precise information about any changes observed during palpation to be judged by clinicians or AI whereas judgement based on human palpation can be subjective.

Acknowledgements

This work was partially supported by Centre for Doctoral Training, United Kingdom (CDT) in Agri-Food Robotics (AgriFoRwArdS) Grant reference: EP/S023917/1; Lincoln Agri-Robotics (LAR) funded by Research England; and by ARTEMIS project funded by Cancer Research UK C24524/A300038.

References

Akgun, B.; Cakmak, M.; Jiang, K.; and Thomaz, A. L. 2012. Keyframe-based learning from demonstration. *International Journal of Social Robotics*, 4(4): 343–355.

Bishop, C. M. 2006. Linear Models for Regression. In Jordan, M.; Kleinberg, J.; and Schölkopf, B., eds., *Pattern recognition and machine learning*, chapter 3, 137–147. New York: Springer.

Deepthi, G. L. 2017. Breast screening. <https://www.slideshare.net/LAKSHMIDEEPHTHIGEDELA/breast-screening-81998153>. Accessed: 2021-08-28.

Ferlay, J.; Ervik, M.; Lam, F.; Colombet, M.; Mery, L.; and Piñeros, M. 2021. Global Cancer Observatory: Cancer Today. <https://www.who.int/news-room/fact-sheets/detail/cancer>. Accessed: 2021-08-30.

Finn, C.; and Levine, S. 2017. Deep visual foresight for planning robot motion. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2786–2793. IEEE.

Ghalamzan E., A.; and Ragaglia, M. 2018. Robot learning from demonstrations: Emulation learning in environments with moving obstacles. *Robotics and autonomous systems*, 101: 45–56.

Girgin, H.; Pignat, E.; Jaquier, N.; and Calinon, S. 2020. Active improvement of control policies with Bayesian Gaussian mixture model. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 5395–5401. IEEE.

Herzig, N.; Maiolino, P.; Iida, F.; and Nanayakkara, T. 2018. A variable stiffness robotic probe for soft tissue palpation. *IEEE Robotics and Automation Letters*, 3(2): 1168–1175.

Houghton, N.; Jenkinson, G. P.; van Zalk, N.; Waller, J.; Bello, F.; and Tzemanaki, A. 2021. Acceptability of Automated Robotic Clinical Breast Examination. Forthcoming.

Jaquier, N.; Ginsbourger, D.; and Calinon, S. 2019. Learning from demonstration with model-based Gaussian process. arXiv:1910.05005.

Keshavarz, M.; and Mojra, A. 2015. Dynamic modelling of breast tissue with application of model reference adaptive system identification technique based on clinical robot-assisted palpation. *Journal of the mechanical behavior of biomedical materials*, 51: 269–278.

Kobayashi, Y.; Suzuki, M.; Kato, A.; Konishi, K.; Hashizume, M.; and Fujie, M. G. 2009. A robotic palpation-based needle insertion method for diagnostic biopsy and treatment of breast cancer. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 5534–5539. IEEE.

Kösters, J.; and Göttsche, P. 2003. Regular self-examination or clinical examination for early detection of breast cancer. *Cochrane database of systematic reviews (Online)*, 2: CD003373.

Levine, S.; Finn, C.; Darrell, T.; and Abbeel, P. 2016. End-to-end training of deep visuomotor policies. *The Journal of Machine Learning Research*, 17(1): 1334–1373.

Levine, S.; Pastor, P.; Krizhevsky, A.; Ibarz, J.; and Quillen, D. 2018. Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *The International Journal of Robotics Research*, 37(4-5): 421–436.

²<https://bit.ly/3GFIsc3>

- Levine, S.; Popovic, Z.; and Koltun, V. 2011. Nonlinear inverse reinforcement learning with gaussian processes. *Advances in neural information processing systems*, 24: 19–27.
- Mahoney, L.; and Csima, A. 1982. Efficiency of palpation in clinical detection of breast cancer. *Canadian Medical Association Journal*, 127(8): 729.
- Matsubara, T.; Hyon, S.-H.; and Morimoto, J. 2011. Learning parametric dynamic movement primitives from multiple demonstrations. *Neural networks*, 24(5): 493–500.
- Murali, M. E.; and Crabtree, K. 1992. Comparison of two breast self-examination palpation techniques. *Cancer nursing*, 15(4): 276–282.
- Nagahama, K.; and Yamazaki, K. 2019. Learning from Demonstration Based on a Mechanism to Utilize an Object's Invisibility. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 6120–6127. IEEE.
- Nichols, K. A.; and Okamura, A. M. 2015. Methods to segment hard inclusions in soft tissue during autonomous robotic palpation. *IEEE Transactions on Robotics*, 31(2): 344–354.
- Paraschos, A.; Daniel, C.; Peters, J.; and Neumann, G. 2018. Using probabilistic movement primitives in robotics. *Autonomous Robots*, 42(3): 529–551.
- Paraschos, A.; Daniel, C.; Peters, J. R.; and Neumann, G. 2013. Probabilistic Movement Primitives. In Burges, C. J. C.; Bottou, L.; Welling, M.; Ghahramani, Z.; and Weinberger, K. Q., eds., *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc.
- Pervez, A.; Mao, Y.; and Lee, D. 2017. Learning deep movement primitives using convolutional neural networks. In *2017 IEEE-RAS 17th international conference on humanoid robotics (Humanoids)*, 191–197. IEEE.
- Provencher, L.; Hogue, J.; Desbiens, C.; Poirier, B.; Poirier, E.; Boudreau, D.; Joyal, M.; Diorio, C.; Duchesne, N.; and Chiquette, J. 2016. Is clinical breast examination important for breast cancer detection? *Current Oncology*, 23(4): 332–339.
- Qi, C. R.; Su, H.; Mo, K.; and Guibas, L. J. 2017. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. arXiv:1612.00593.
- Rahmatizadeh, R.; Abolghasemi, P.; Bölöni, L.; and Levine, S. 2018. Vision-based multi-task manipulation for inexpensive robots using end-to-end learning from demonstration. In *2018 IEEE international conference on robotics and automation (ICRA)*, 3758–3765. IEEE.
- Rana, M.; Mukadam, M.; Ahmadzadeh, S. R.; Chernova, S.; and Boots, B. 2018. Towards robust skill generalization: Unifying learning from demonstration and motion planning. In *Intelligent robots and systems*.
- Ridge, B.; Gams, A.; Morimoto, J.; Ude, A.; et al. 2020. Training of deep neural networks for the generation of dynamic movement primitives. *Neural Networks*, 127: 121–131.
- Rueckert, E.; Mundo, J.; Paraschos, A.; Peters, J.; and Neumann, G. 2015. Extracting low-dimensional control variables for movement primitives. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 1511–1518. IEEE.
- Saslow, D.; Hannan, J.; Osuch, J.; Alciati, M. H.; Baines, C.; Barton, M.; Bobo, J. K.; Coleman, C.; Dolan, M.; Gaumer, G.; et al. 2004. Clinical breast examination: practical recommendations for optimizing performance and reporting. *CA: a cancer journal for clinicians*, 54(6): 327–344.
- Schaal, S. 2006. Dynamic movement primitives—a framework for motor control in humans and humanoid robotics. In *Adaptive motion of animals and machines*, 261–280. Springer.
- Schneider, M.; and Ertel, W. 2010. Robot learning by demonstration with local gaussian process regression. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 255–260. IEEE.
- Scimeca, L.; Maiolino, P.; Bray, E.; and Iida, F. 2020. Structuring of tactile sensory information for category formation in robotics palpation. *Autonomous Robots*, 44(8): 1377–1393.
- Sung, H.; Ferlay, J.; Siegel, R. L.; Laversanne, M.; Soerjomataram, I.; Jemal, A.; and Bray, F. 2021. Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: a cancer journal for clinicians*, 71(3): 209–249.
- Ude, A.; Gams, A.; Asfour, T.; and Morimoto, J. 2010. Task-specific generalization of discrete and periodic dynamic movement primitives. *IEEE Transactions on Robotics*, 26(5): 800–815.
- Yang, R.-J.; Huang, L.-H.; Hsieh, Y.-S.; Chung, U.-L.; Huang, C.-S.; and Bih, H.-D. 2010. Motivations and reasons for women attending a breast self-examination training program: a qualitative study. *BMC women's health*, 10(1): 1–11.