‘Feel the Painting’: a Clinician-Friendly Approach to Programming Planar Force Fields for Haptic Devices

Paolo Tommasino, Asif Hussain, Aamani Budhota, Charmayne ML Hughes,Wayne Dailey, Domenico Campolo
Robotics Research Centre, School of Mechanical and Aerospace Engineering Nanyang Technological University, Singapore
Email: paolo001,d.campolo@ntu.edu.sg

***Abstract*—Haptic force fields are widely used in studies on motor adaptation, motor retention, and motor recovery in both healthy and impaired subjects. In the main paradigm the hand is guided or perturbed along specific paths or channels in order to investigate different aspects underlying the human motor control. Programming such fields for complex *haptic environments* can be very challenging and is often not feasible for clinicians and therapists. The aim of this paper is to introduce a more intuitive and clinician-friendly programming method capable of transforming a 2D drawing (stored as an image) into a haptic environment or planar force field. By considering the image intensity as a position-dependent potential field, the energy function is approximated through locally weighted projection regression (LWPR). Robot forces are then computed through the gradient of the regressed potential. The proposed method is validated with a two degrees-of-freedom planar manipulandum, the H-Man, and a preliminary shape recognition experiment involving blindfolded healthy subjects.**

I. Introduction

In recent decades, upper extremity robots have received increasing attention as a valuable means to understand motor adaptation, learning, and retention in both healthy and in impaired subjects. Despite the increasing number of devices developed so far [1], remarkable little focus has been applied to make the programming and control of such devices more intuitive and therapist-friendly.

Commercially available robots usually come with a software interface with settings that allow for the easy adjustment of robotic assistance level, gaming/graphical user interface, number of trials etc. However, there is very little flexibility in terms of the types of motions that the robot and patient can perform, with most interfaces limiting motion to point-to-point straight-line trajectories. This limitation is a particular challenge for therapists who lack sufficient engineering and programming expertise required to programme more complex and time consuming robotic force fields.

Typically, robotic force fields are programmed to reduce or enhance movement errors. Robotic force fields are said to be *assistive* or *convergent* whenever the robot acts in order to *reduce* a given error measure (usually the lateral deviation from a nominal path), while *divergent* force fields are those which amplify motor errors [8], [10]. The most commonly used force fields are *velocity-dependent* (or viscous) fields in which the robotic action response is proportional to the subject hand speed or *position-dependent* force fields by which the robot reacts based on a position-dependent error measure. While both fields have been employed to examine motor control in healthy and impaired subjects, given that this project is still in the preliminary phase, this paper will address only the intuitive and clinician-friendly robot programming of position-dependent force fields.

A. Force fields for impaired and healthy motor control

While planar point-to-point reaching movements performed by healthy subjects are characterized by smooth and roughly straight-line motions [2], individuals with motor and proprioceptive impairments make slow reaching movements that are characterised by jerky velocity profiles and high end-point variability [3].

Assistive force fields have been used to limit the aforementioned issues in impaired subjects. In [4] a *canyon-like* energy function was used to stiffen up the robot end-effector against lateral deviations from straight-line paths connecting the starting position with eight peripheral targets. Studies that employed convergent force fields to examine motor adaptation in healthy subjects have shown that the introduction of an *haptic channel* subsequent to motor adaptation to viscous fields can induce a prolonged memory retention of the acquired force field [5]. The level of channel compliance has also been shown to have a significant effect on the amount of induced adaptation [6].

Because of their intrinsic nature, assistive force fields often lead to low engagement and effort levels; both healthy and impaired subjects tend to react passively to robot motions and guidance [6], [9]. Error-enhancing or *divergent* force fields have been proposed as a more stimulating alternative as prediction errors (between the expected and executed motions) are thought to trigger motor adaptation and learning processes [7]–[10].

B. Programming Haptic Force Fields

Most of the aforementioned proposed robotic therapy solutions require precomputed trajectories along which the robot end-effector can be modulated through an impedance controller in order to enhance or reduce motor errors [4],

978-1-4799-1808-9/15/$31.00 ©2015 IEEE

[8], [9]. Such an approach however, can be very tedious for paths which do not have a direct mathematical representation (such as a maze or the environment shown in Fig.1). Also, implementing new paths require time-consuming robot reprogramming activities which are infeasible for most individuals with medical backgrounds.

More intuitive robot programming methods have been proposed to assist hand-writing and drawing skills. In [11], desired robot trajectories were first recorded from an experienced user, and then a proportional-derivative controller was used to provide motion guidance in unskilled users. A solution to generate robot trajectories from images was proposed in [12] where computer vision techniques (such as edge detection and image segmentation) were used to extract desired robot paths which were tracked by an impedance controller to assist the users motion. Although these methods are more user-friendly, and can extract even complex robot paths, they rely on impedance controllers which cannot render textured landscapes made of both convergent and divergent fields (such as the environment shown in Fig.1).

Image-based force rendering that avoids tracking specific robot paths have been proposed in the context of haptic texture rendering. In [13], a scaled version of the image gradient, computed through the finite difference method of consecutive pixels, was used as desired force for the haptic display. In [14], an $M \times M$ mask, centred on the end-effector of the haptic display, was used to compute an average gradient of the corresponding pixel. However, the image resolution is critical for these methods: when the finite-difference method is directly applied to the pixel intensity, the resultant forces can be discontinuous and bumpy, especially for low resolution images. Conversely, the computational burden of run-time pixel processing for high resolution images makes it difficult to perform real-time robot control. An image-based programming method for robotics rehabilitation has been proposed in [15] where a binary image (black and white) was used to generate a custom 2D maze. At each time step, the algorithm compared the coordinates of the subject hand with the closest pixel in the image and a corresponding haptic feedback was generated. A limitation of this work is that although it uses a 2D drawing does not allow for the generation of textured force fields, and requires a suitable game engine for collision detection.

Rather than directly computing the gradient from the image, our method regresses the image intensity through non-linear function approximation and therefore has several advantages when compared to the previous solutions. First, the advantage of LWPR is that the regressed model will have a computational burden depending on the complexity of the image rather than on image resolution. In other words, more local models (see sec. II-A) will be allocated in those regions of the image with several details while only few local models will be allocated in those regions with constant intensity levels. Learning the image intensity map (as a potential function) has the computational benefit of dealing only with a scalar function approximator rather than with a vector-valued one required to directly learn the image gradient. Furthermore, because the regressed function is a smoothed version of the input image the estimated gradient will be smoother than that directly estimated from the image.

II. METHODS

The input to our algorithm is a gray-scale image which contains the drawing of the *haptic environment* that the experimenter or clinician/caregiver wishes to render. With a gray-scale image there is the possibility of choosing among 256 intensity levels ranging from 0 (black) to 255 (white). We assume that the image intensity represents a position-dependent potential field that describes the energy landscape of the robot end-effector. For instance, a convergent haptic channel can be painted by drawing with a dark colour ink over a lighter background. Similarly, a divergent force field can be painted by using a lighter coloured ink over a darker background. The width of the channel can then be modulated by using different stroke sizes. By using different combinations of intensity levels it is therefore possible to segment the robot workspace into *attractive* and repulsive areas (see Fig. 1).

A. Learning the potential field with LWPR

In theory, any function approximator can be used to regress the energy function of the potential field. In this paper we propose the use of locally weighted projection regression (LWPR) [16], [17], a non-linear function approximator that has been used extensively for robot control. LWPR is an efficient algorithm which achieves nonlinear function approximation by using a weighted sum of locally linear models. Such nonparametric regression technique learns rapidly with second order learning methods using statistically sound stochastic cross validation [16], [17]. It has a computational complexity that is linear in the number of inputs, it automatically allocates and tunes the linear models on a *where-needed* basis. As such, the input space is efficiently covered (more and narrower receptive fields are allocated where the input space has a higher curvature), and the gradient of the learned model can be derived analytically and computed efficiently [17].

Let $\boldsymbol{p} = [x, y]^T$ be a point of the robot workspace. The LWPR output (or the estimated image intensity) at $\boldsymbol{p}$ is computed as the combination of K weighted locally linear models normalized by the sum of all weights [16]:

$$\hat{\Phi}(\boldsymbol{p}) = \frac{\sum_{k=1}^{K} \boldsymbol{w}_k(\boldsymbol{p}) \bar{\Phi}_k(\boldsymbol{p})}{\sum_{k=1}^{K} \boldsymbol{w}_k(\boldsymbol{p})} \tag{1}$$

where $\bar{\Phi}_k(\boldsymbol{p}) = b_k^0 + \boldsymbol{b}_k{}^T(\boldsymbol{p} - \boldsymbol{c}_k)$ is the output of the k-th linear model and b_k^0 and $\boldsymbol{b}_k{}^T$ its regression parameters representing the offset and the slope respectively.

For Gaussian weights (or receptive fields), $\boldsymbol{w}_k(\boldsymbol{p}) = exp(-0.5(\boldsymbol{p} - \boldsymbol{c}_k)^T D_k(\boldsymbol{p} - \boldsymbol{c}_k))$ where D_k is a positive definite distance matrix that regulates the region of influence

of the k-th linear model, and the vector $\boldsymbol{c}_k$ is the position of k-th linear over the robot workspace. The goal of the training process is to tune D_k and the regression parameters in order to reduce the error between the predicted output $\hat{\Phi}(\boldsymbol{p})$ and the target image-intensity values $\Phi(\boldsymbol{p})$.

Before training, the input image is smoothed with a Gaussian filter to reduce the roughness of low-resolution images. For an $N \times N$ image, there are $T = N^2$ pixels or training samples. The image intensity levels Φ_m with $m = 1, ..., T$, represent the desired output, and are rescaled in the range $[0, 1]$ to facilitate the LWPR training [17]. The discrete image coordinates (i, j) of the m-th pixel are linearly mapped into robot workspace (or Cartesian) coordinates $\boldsymbol{p}_m = (x_m, y_m)$ with $m = 1, \ldots, T$. The T remapped pixel coordinates, together with the T normalized pixel intensity Φ_m, are used as training samples for the LWPR model: *lwpr_update*$(\boldsymbol{p}_m, \Phi_m)$ [1]. Several training cycles are used to tune the LWPR model. A training cycle is characterised by a particular sequence of input-output pair $(\boldsymbol{p}_m, \Phi_m))$ which is then randomized for the next cycle. The training is terminated when the normalized mean square error (Eq. 2) is less than 0.01, or when 150 training cycles have been performed.

$$e = \frac{\sum_{i=1}^{T}(\hat{\Phi}_i - \Phi_i)^2}{T}\mathrm{Var}(\Phi) \tag{2}$$

where $\mathrm{Var}(\Phi)$ is the variance of the normalized image intensity values.

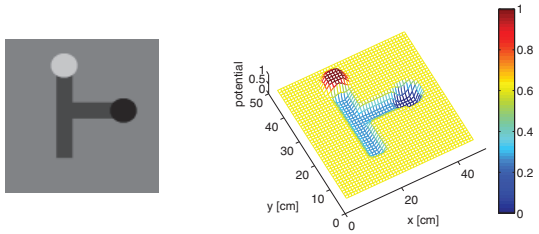
1) Parameters tuning: LWPR performance depends strictly on the parameter setting chosen for the training [17]. For our particular approximation problem we found out that the main parameters affecting the performance were: the initial width of the recptive fields (*init_D*) and the weight activation threshold (*w_gen*) below which a new receptive field is allocated. For the images tested in this work, we set *init_D*$= 5I$ where I is the 2×2 identity matrix, *w_gen*$= .6$ and *diag_only*$= 0$ so as to adapt the receptive field width over the course of learning. This setting represented a good trade-off between the force field smoothness and computational burden. For both images tested in this work (see sec. III) made up of 10^4 pixels, the LWPR algorithm allocated less than 2500 local models. On a PC with a 3.40GHz Intel(R) Core(TM) i5 Dual-core with 4 GB of RAM the average computational time for the gradient estimation took approximately 0.2 ms, and therefore suitable for real-time applications.

B. Gradient Estimation and Robot Control

The gradient of the LWPR model (1) can be analytically computed as shown in [17] with a significant computational benefit when compared to finite difference method [19]. For a given robot end-effector position $\boldsymbol{p}$, we compute the image-based force field $\boldsymbol{F}(\boldsymbol{p})$ through the gradient of the regressed potential:

$$\boldsymbol{F}(\boldsymbol{p}) = -\alpha \nabla \hat{\Phi}(\boldsymbol{p}) \tag{3}$$

[1] *lwpr_update* is the training function of the LWPR library [17], [18].



(a) (b)

Fig. 1. (a) Image representation of an *haptic environment* characterized by a T-shaped convergent channel, an attractive target (black circle) and a repellent target (light gray circle). (b) The corresponding energy levels of the potential field.

where $\nabla \hat{\Phi}(\boldsymbol{p})$ is computed with the LWPR function *lwpr_predict_J*. [17], [18] The absolute value of the scalar α modulates the strength of the force field as the potential function can only take on values between 0 and 1. For the same image, a negative sign of α would transform repellent areas into attractive ones, and vice versa.

For a robot having n degrees-of-freedom whose end-effector moves into a an m-dimensional task-space, the $m \times n$ Jacobian matrix J maps joints velocities $(\dot{\boldsymbol{q}})$ into end-effector velocities: $\dot{\boldsymbol{x}} = J\dot{\boldsymbol{q}}$. The transposed Jacobian matrix can also be used to map task-space forces into joint-space torques [22].Therefore, once the task-space forces $(\boldsymbol{F}(\boldsymbol{p}))$ have been estimated via the LWPR model, the corresponding motor torques can be computed as:

$$\boldsymbol{\tau} = J^T(\boldsymbol{F}(\boldsymbol{p}) - B\dot{\boldsymbol{p}}) \tag{4}$$

where the additional damping force $-B\dot{\boldsymbol{p}}$ is used for stabilization purposes.

III. RESULTS

In this section we delineate the preliminary results for two assistive or convergent force fields made of a *SPIRAL* and an *ARC* respectively (see Fig2(a) and (e)). For the ARC image, we could have drawn a black arc on a gray background. However we decided to represent this environment with a black circle with a superimposed white ellipse to validate the possibility of generating textured force fields composed of different gray-scale levels. Furthermore, it should be noted that rendering the ARC environment with conventional robot programming methods can be quite challenging and time consuming.

These haptic shapes were selected in order to validate the proposed controller along curved paths which are generally difficult to implement with an image-based programming method, but also have a long history in the field of human motor control [20].

The 100×100 pixels PNG images (Fig.2(a) and (e)) were generated using the open source software Inkscape. The spiral was drawn with a 5 pixels stroke size while the circle with a 8 pixels stroke size. The images were smoothed with a 2×2 Gaussian filter (see Fig.2(b)and(f)) and the

pixels were remapped onto a $25 \times 25[cm]$ workspace grid. After training, LWPR generalization capabilities were tested

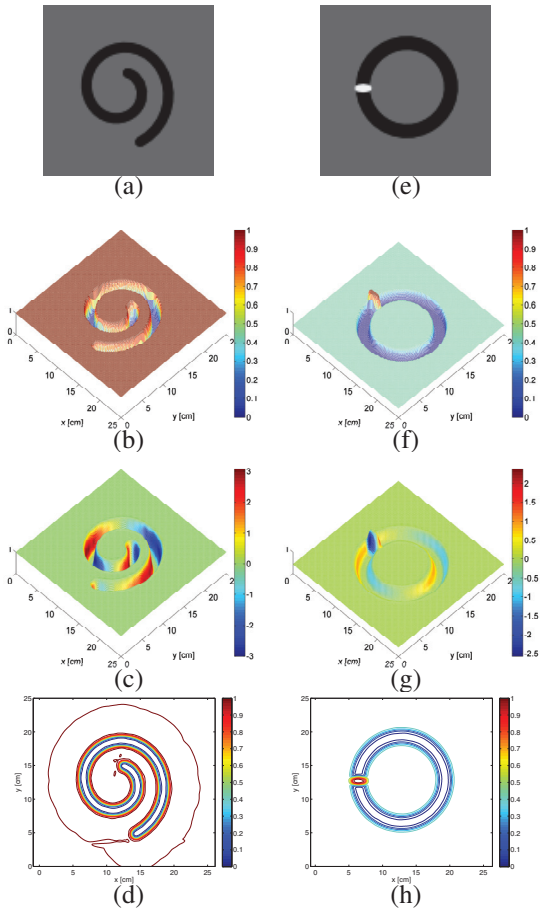


Fig. 2. LWPR predictions. (a) and (e) Input images of 100×100 pixels. (b) and (f) The input images are smoothed with a Gaussian filter, the intensity levels rescaled in the range $[0, 1]$, and the pixel coordinates are remapped into Cartesian coordinates. (c) and (g) LWPR regressed potential and estimated gradients (color bars) onto a 400×400 input grid. (d) and (h) Contour plots for the LWPR regressed potentials.

onto a 400×400 grid with samples uniformly distributed in the range 0 to 25 cm. Regressed potentials and estimated gradients are shown in Fig.2(c) and (g) and, Fig.2(d) and (h) show the contour plots of the regressed potential functions. For both environments, the LWPR models capture both the shape and the energy levels of the desired potential functions (normalized pixel intensities). For the SPIRAL environment the haptic channel has a gradient ranging from -3 to 3 N (compare the color bar in in Fig.2(c)). The ARC environment on the other hand, has a haptic channel with a gradient ranging from -.5 to .5 N that increases from -2.5 to 2.5 N in the proximity of the white ellipse (compare the color bar in in Fig.2(c)). Thus, for the same α level (see Eq.3), the haptic channel of the ARC will be less stiff than that of the SPIRAL.

A. *Shape recognition by blind-folded subjects*

To validate the quality of the haptic rendering using our approach we conducted a preliminary experiment with the H-Man, a two degrees-of-freedom planar robotic manipulandum designed for the rehabilitation of the upper extremity [6], [21]. The aim of the experiment is to examine whether blindfolded individuals are able to *feel* the force field, as well as to locate and distinguish the two haptic shapes. For both the environments we set $B = 20I$ (in Eq. 4), where I is the 2×2 identity matrix, to add an isotropic damping action to the estimated forces.

Two healthy participants naive to the purpose of the experiment took part in the study. Subjects were asked to recognize a shape consisting of a piece of non-intersecting 2D curve with a starting point and an ending point, but no additional information was given regarding the type of shape. Subjects were asked to first explore the robot workspace, locate the position of the starting and ending points within the shape, and to provide a verbal statement about the type of shape once they felt confident with the shape they were following. The scalar α (see Eq.3) was kept constant and equal to 7 for the SPIRAL environment and 10 for the ARC (as its haptic channel is weaker than the SPIRAL one [compare colorbar in Fig.2(c) and (g)]).

At the beginning of each test the H-Man end-effector was initialized at the bottom left corner of its workspace. The two force fields were presented in opposite order for the two subjects; subject A started with the ARC and subject B with the SPIRAL (Fig.3 (a) and (c) respectively). For subject A (see Fig.3(a)) we observed a wider exploration of the workspace for the ARC environment (that was also the participant first exposure to the experiment). By comparison (see Fig.3(c)), the subject was able to quickly locate the spiral and its hand movements were mainly confined along one border of the haptic channel of the SPIRAL environment. Subject B, who was first exposed to the SPIRAL (see Fig.3(b)), quickly located the spiral by moving toward the center of the workspace and the subject hand movements were constrained inside spiral haptic channel until the end of the experiment. By comparison (see Fig.3(d)), subject B moved in and out from the channel however with hand movements mainly confined around the shape. Fig.4(a) shows the distribution of the hand velocities for both subjects and environments. At each time step the norm of the hand velocity was computed and the *ksdensity* Matlab function was used for the plot. Both subjects explored at a lower speed during the first exposure to the experiment (participant A-ARC and participant B-SPIRAL) when compared to the second exposition (participant A-SPIRAL and participant B-ARC), suggesting more confident behaviour likely due to increased task familiarity. Overall, subject B explored with a higher hand speed than subject A. The force distribution as in Eq.4 delivered by the H-Man are shown in Fig.4(b). By moving at higher speed, subject B experienced a wider range of forces due to the damping component in Eq.4.

Both subjects guessed 'spiral' for the SPIRAL environment, while subject B reported 'backward C' and subject A 'circle' for the ARC environment. Subject A reported that the starting and ending point were located in the same position, indicating that his guess was due to proprioceptive feeling. The *time-to-guess* is shown in Fig.5. Due to a wider workspace exploration and to lower speed movements, subject A took approximately 3.5 minutes to guess its first shape (ARC), while subject B took less than 2 minutes to guess the first shape (SPIRAL). A series of back-and-forth *confident* movements after the shape was correctly guessed are shown in Fig.6.

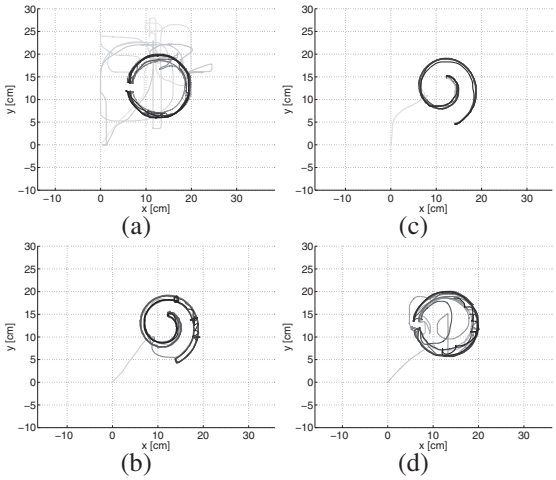


Fig. 3. Hand paths of the two blindfolded subjects during the shape recognition test. (a) and (c) Subject A performing the ARC and the SPIRAL test respectively. (b) and (d) Subject B performing the SPIRAL and the ARC test respectively. Most recent paths are in darker colors.

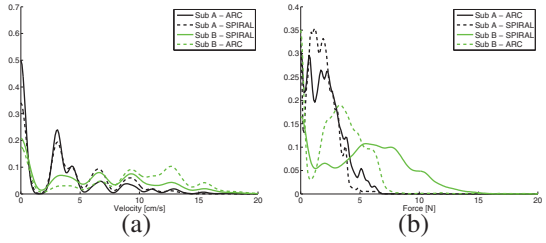


Fig. 4. (a) Norm of the end-effector velocities distributions for each subject and for each haptic environment. It is possible to notice that Subject A uses a lower speed exploration strategy when compared to Subject B. (b) Norm of the task-space robot forces ($\boldsymbol{F}(\boldsymbol{p}) - B\dot{\boldsymbol{p}}$) distribution. Because subject B moves faster than subject A, he experiences an higher level of robot forces due to stabilizing damping force. The Matlab function *ksdensity* was used for both plots.

IV. Conclusion

This paper proposes a novel method to convert 2D sketches stored as gray-scale images into planar force fields which are commonly used in robotics rehabilitation and haptics. At the core of the algorithm is the assumption that the image intensity levels represent an energy function distributed over the robot workspace. The energy function is regressed

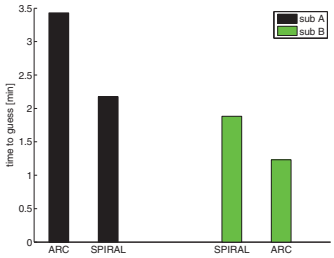


Fig. 5. Time to guess (in minutes) the haptic shape.

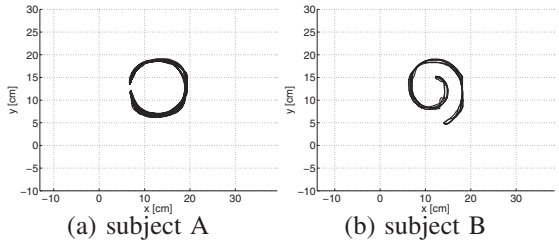


Fig. 6. Hand paths of the two blindfolded subjects after familiarizing with the haptic shapes.

through LWPR so that the gradient of the regressed model can be efficiently computed to generate the robot forces. The task-space formulation of the proposed solution makes it possible to generate planar force fields with any robotic platform. This new approach is conceived as an intuitive robot programming method for use by therapists and caregivers with no or minimal programming experience, but who are directly responsible for carrying out therapy regimes with impaired or elderly participants. Although in this paper we reported experiments that do not require visual feedback, it should be noted that the input image can be easily imported into conventional game engines used in rehabilitation robotics and haptics. . The advantages of this feature is that it reduces (or potentially eliminates) the additional programming time required to develop a gaming environment.

To validate the quality of the rendered force field we devised a preliminary experiment involving a haptic shape recognition task. Both blindfolded participants were successful in their shape estimations. As such, the proposed method provides a promising avenue for future research, especially in the field of active touch and sensorimotor training where the type of movements are, for the most part, confined to point-to-point reaching movements. For instance, future works could address differences in terms of explorations patterns and shape perception between convergent and divergent channels. In addition, it would certainly be worthwhile examining differences in motor adaptation and explorations strategies between abstract and conventional shapes (e.g. geometric shapes, numbers, letters, etc.).

At the current stage some limitations of our implementation are worth noting: 1) the use of conventional software for image editing does not provide an intuitive or transparent means for users to understand how the drawing and its details

(such as channels width, target size...) would scale up in the robot workspace; 2) LWPR training parameters are currently manually tuned by a trial-and-error process. Future works will be devoted to the realization of an integrated software interface which will allow the image painting over a grid that is already scaled to the robot workspace. Last, the automatic tuning of the LWPR parameters based on the extraction of the finer details present in the drawing will be addressed. The acceptability and easy of use of such interface by non-expert users will be quantitatively evaluated.

ACKNOWLEDGMENT

This work was partly supported by the H-Man project (NMRC/BnB/0006b/2013), Ministry of Health, and by the Academic Research Fund (AcRF) Tier1 (RG 50/11), Ministry of Education, Singapore.

REFERENCES

[1] Loureiro, R. C., Harwin, W. S., Nagai, K., Johnson, M. (2011). *Advances in upper limb stroke rehabilitation: a technology push.* Medical & biological engineering & computing, 49(10), 1103-1118.

[2] Morasso, P. (1981). *Spatial control of arm movements.* Experimental brain research, 42(2), 223-227.

[3] Hughes CML, Tommasino P, Budhota A, Campolo D (2015). *Upper extremity proprioception in healthy aging and stroke populations, and the effects of therapist- and robot-based rehabilitation therapies on proprioceptive function.* Front. Hum. Neurosci. 9:120. doi: 10.3389/fnhum.2015.00120

[4] Krebs, H. I., Palazzolo, J. J., Dipietro, L., Ferraro, M., Krol, J., Rannekleiv, K., Hogan, N. (2003). *Rehabilitation robotics: Performance-based progressive robot-assisted therapy.* Autonomous Robots, 15(1), 7-20.

[5] Scheidt, R. A., Reinkensmeyer, D. J., Conditt, M. A., Rymer, W. Z., Mussa-Ivaldi, F. A. (2000). *Persistence of motor adaptation during constrained, multi-joint, arm movements.* Journal of Neurophysiology, 84(2), 853-862.

[6] P. Tommasino, A. Melendez-Calderon, E. Burdet, D. Campolo. *Motor adaptation with passive machines: A first study on the effect of real and virtual stiffness.* Computer Methods and Programs in Biomedicine, Volume 116, Issue 2, September 2014

[7] Shadmehr, R., Mussa-Ivaldi, F. A. (1994). *Adaptive representation of dynamics during learning of a motor task.* The Journal of Neuroscience, 14(5), 3208-3224.

[8] Cesqui, B., Aliboni, S., Mazzoleni, S., Carrozza, M. C., Posteraro, F., Micera, S. (2008, October). *On the use of divergent force fields in robot-mediated neurorehabilitation.* In Biomedical Robotics and Biomechatronics, 2008. BioRob 2008. 2nd IEEE RAS & EMBS International Conference on (pp. 854-861). IEEE.

[9] Tropea, P., Cesqui, B., Monaco, V. I. T. O., Aliboni, S. A. R. A., Posteraro, F., Micera, S. (2013). *Effects of the Alternate Combination of Error-Enhancing and Active Assistive Robot-Mediated Treatments on Stroke Patients.* Translational Engineering in Health and Medicine, IEEE Journal of, 1, 2100109-2100109.

[10] Patton, J. L., Stoykov, M. E., Kovic, M., Mussa-Ivaldi, F. A. (2006). *Evaluation of robotic training forces that either enhance or reduce error in chronic hemiparetic stroke survivors.* Experimental Brain Research, 168(3), 368-383.

[11] Teo, C. L., Burdet, E., Lim, H. P. (2002). *A robotic teacher of Chinese handwriting.* In Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2002. HAPTICS 2002. Proceedings. 10th Symposium on (pp. 335-341). IEEE.

[12] Portillo, O., Avizzano, C. A., Raspolli, M., Bergamasco, M. (2005, August). *Haptic desktop for assisted handwriting and drawing.* In Robot and Human Interactive Communication, 2005. ROMAN 2005. IEEE International Workshop on (pp. 512-517). IEEE.

[13] Minsky, M. D. R. (1995). *Computational haptics: the sandpaper system for synthesizing texture for a force-feedback display* (Doctoral dissertation, Massachusetts Institute of Technology).

[14] Vasudevan, H., Manivannan, M. (2008, March). *Tangible images: runtime generation of haptic textures from images.* In Haptic interfaces for virtual environment and teleoperator systems, 2008. haptics 2008. symposium on (pp. 357-360). IEEE

[15] Sivak, M., Unluhisarcikli, O., Weinberg, B., Mirelman-Harari, A., Bonato, P., Mavroidis, C. (2010, March). *Haptic system for hand rehabilitation integrating an interactive game with an advanced robotic device.* In Haptics Symposium, 2010 IEEE (pp. 475-481). IEEE.

[16] Vijayakumar, S., Schaal, S. (2000, June). *Locally weighted projection regression: Incremental real time learning in high dimensional space.* In Proceedings of the Seventeenth International Conference on Machine Learning (pp. 1079-1086). Morgan Kaufmann Publishers Inc.

[17] Klanke, S., Vijayakumar, S., Schaal, S. (2008). *A library for locally weighted projection regression.* The Journal of Machine Learning Research, 9, 623-626.

[18] Klanke, S., Vijayakumar, S., Schaal, S. (2008). *A library for locally weighted projection regression - Supplementary Materials.* http://wcms.inf.ed.ac.uk/ipab/slmc/research/software-lwpr.

[19] Mitrovic, D., Klanke, S., Vijayakumar, S. (2010). *Adaptive optimal feedback control with learned internal dynamics models.* In From Motor Learning to Interaction Learning in Robots (pp. 65-84). Springer Berlin Heidelberg.

[20] Lacquaniti, F., Terzuolo, C., Viviani, P. (1983). *The law relating the kinematic and figural aspects of drawing movements.* Acta psychologica, 54(1), 115-130.

[21] Campolo, D., Tommasino, P., Gamage, K., Klein, J., Hughes, C. M., Masia, L. (2014). *H-Man: A planar, H-shape cabled differential robotic manipulandum for experiments on human motor control.* Journal of neuroscience methods, 235, 285-297.

[22] Murray, R. M., Li, Z., Sastry, S. S., Sastry, S. S. (1994). A mathematical introduction to robotic manipulation. CRC press.