How Do You Feel Today? Emotional Expressive Robot Motion

Chen-Lung Lu

***Abstract*—Expressing inner state via motions is one characteristic of robots that other machines can not have in human-computer interaction. Companies and researcher have long been searching for method for robots to act more human-like. In this project, I would like the robot to learn to do emotional expressive motions. In particular, the robot will learn to expressive happy motion. To do so, I tried to find the emotion prediction function in which given a set of motion parameters, the function will predict the emotion score. The project deployed the framework followed the one proposed by [1] but with other kind of robot and designed motion parameters based on observation of previous literature. I first sampled a motion dataset from designed parameters and labeled the motions using a tournament with emotion score inference. The emotional motion is then generated from the learned function from the data. The generated motions were evaluated through crowdsourcing and showed people do think the robot is happier while the emotion score is higher. The motions were also deployed on the real robot to demonstrate the physical feasibility of these generated motion. Finally, a discussion is made upon the result of the project and pointed out way to improve the result and future research direction.**

The code and result can be found on https://github.com/eric565648/robotics1_project.

A short video of the project can be found at https://youtu.be/G78Iz2lNfrQ

***Index Terms*—Expressive Robots, Mobile Manipulator, Robot Arm, Crowd-Sourcing**

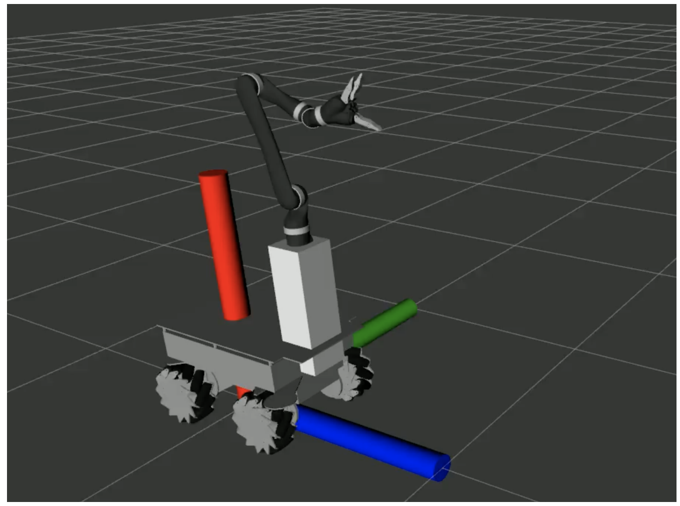

Fig. 1: Oarbot is being ”happy”. The robot is expressing ”happy” in which was learned from a dataset collected by comparing happiness of motions. The robot platform, Oarbot, used in the project is a 10 DoF mobile manipulator.

I. Introduction

Robots have an advantage that other machines, e.g. smartphones, laptops, can't even imaging when comes to human-computer (or human-robot) interaction is to express the inner states of robots via motions. The way to communicate is also straightforward for human. For example, if a person is busy, they may increase their speed of action. Or, if a person is tired, they may lower their limbs or constantly nodding. Motivated by the ability, the project thrived to allow the robot to express one of the states that mostly belongs to human, the emotions. Specifically, the robot learned to express ”happiness” through its motions.

While lots of robots are dedicated into industrial, manufacturing business, there are also effort devoted to make robots more human-like. Disney [2] has long investigated how motions expressive different feeling in cartoons and animation. Moving these guidelines to real world, Disney research groups works to make the robot more human-like via a great control on the body [3], investigating eye gaze [4] and more [5]. The company even want to make a imaginary character ”Groot” to real life [6]. Disney is not the only one thrived to give life to robots. The famous Kuka dance [7] also express feeling through the industrial robot arm.

The project is targeted to make the robot express the emotion, ”happy”, from motions which was learned from a dataset. The motions were first parametrized by several designed motion parameters. The parameters were designed based on past research on how people perceive different motions from robots or real humans. A motion set is then sampled from the motion parameters. While it's necessary to give a ground truth to each motions, it is not practical to ask how people label a motion with a emotion score. Instead, I used a tournament to compare happiness level between motions and finally gives a ground truth score via Microsoft TrueSkill [8]. The function is then learned using regression with the generated dataset. Having this function, the user can know how much ”happy” a motion is and further inversely designed a motion based on given emotion level. I followed the framework which was proposed by] [1] in this project but with different types of robot and introducing different way to designed motion parameters.

To evaluate the results, I generated five motions with different happiness level and asked how happy do people think about certain motion. The result tells that with a higher emotion score, people do think the robot is expressing happiness. The generated motions is also deployed on the real robot arm to show the motions are achievable with real robots. Finally, a discussion was made upon the pros and cons on such method and the qualitative feedback from the questionnaire and pointed out some future research direction for such project.

II. Literature Review

A. Expressive Robot Motion

This work is a replication of [1]. The work proposed a framework to allow robots to learn from human-label emotional motion sets. The work first parametrized the motions into several degrees of freedom from which sampled a large motion

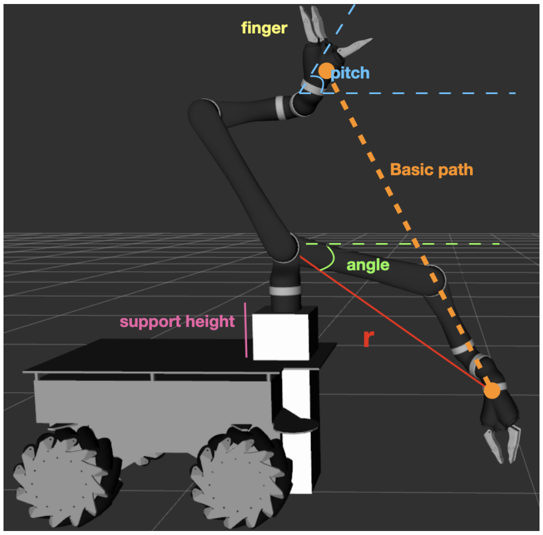

Fig. 2: Starting/Ending Pose parameterization. A basic path (orange dotted line) is then generated from the starting to the ending pose.

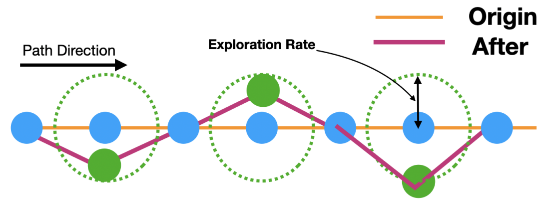

Fig. 3: The exploration rate of the motion. High exploration rate allow the robot to express wiggling motions.

set. To label the motion set with emotional scores, opinions upon which motion has more emotion than another are collected. A ranking system is thus to provide scalar scores to each motion and finally using linear regression to find the relationship of parameters toward emotions. I plan to deploy a similar method in this project; however, I design different motion parameters because an other type of robot is used. In addition, I include the motion parameters associated more with emotions. Furthermore, the code of the original work is not released nor the application. It's a contribution to replicate such work.

There is plenty of works in human-robot interaction studying how motions of robot affect people. [9] explored the affect of abstract motions in which different motions are designed, and users are asked to give opinions on the motions. The work found that even basic robots (e.q. a stick or spring with motors) can express emotions and engage with users. [10] deployed Laban Efforts to the head movements of robots and found that humans can tell the states the robot is presenting. [11] explored the timing of motions in human-robot collaboration. They found the effect of timing on human perception and built a mathematical model that the robot can optimize the timing. Finally, a comprehensive survey was conducted by [12], which includes the generation of whole-body motions with different embodiments such as wheeled, legged, flying systems. The work also includes evaluation methods and future improvements.

III. Expressive Motion Generation

A. Motion Parameterization

From the previous literature, we know that the main factors that affect perceived motions are velocity (speed) and pose of the robot. Therefore I designed the 14 motion parameters where there are 13 continuous and 1 discrete parameters as followed.

1) **Starting Pose: End-Effector Position, Pitch and Finger Position and base height** The pose where the motion started.
2) **Ending Pose: End-Effector Position, Pitch and Finger Position and base height** The pose where the motion ended.
3) **Exploration Rate** The parameter described the probability of the arm exploring to another point while moving along the path.
4) **Arm Speed** The speed of the arm.
5) **Base Speed** The speed of the mobile base.
6) **Velocity Type** The different types of velocity.

The starting and ending pose of the robot is described by the position and pitch of the end-effector, the position of the finger and the height of the base 2. For the position of the end-effector , I used r and θ to described it and restricted the position on the x-z plane to decrease the amount of parameters in ease of learning the mapping. The position and the movement of the finger and the height of the base can potentially express the emotion of a robot. Having the starting and the ending position, I then generated a path from the starting pose to the ending pose which then creates the basic motion of the robot.

The basic path along with the starting and ending pose is not enough to well express emotion. In order to increase the diversity of the motion, the exploration rate parameter is designed 3. After the path is sampled, I chose N waypoint in the path and then created a Gaussian distribution of the position of the waypoint with mean at the current waypoint and the standard deviation as the exploration rate. The step can allow the robot to express wiggling motion when the exploration rate is high. In the actual implementation, I only distributed the waypoint perpendicular to the direction of the path so that the arm will not going back and forth along the path.

Finally, the speed of a motion can also affect the how it is perceived. The speed of the arm and the mobile base were then included in the motion parameters set. I further included the velocity type of the speed where the the path velocity is categorized by six types proposed by [11] which are constant path velocity, path velocity from low to high and high to low, constant path velocity with pause in the middle, path velocity from low to high and high to low and pause in the middle. The path velocity profiles are shown in Fig.4.

B. Motion Synthesizing

Motion synthesizing can be separated to two the motion of the base and the motion of the arm because of how the motion

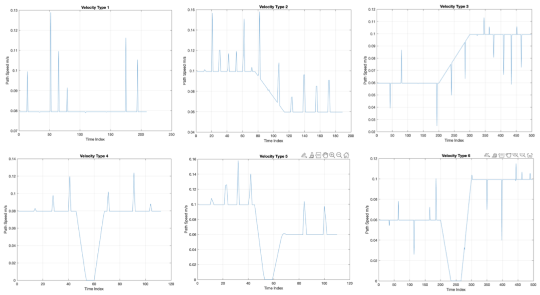

Fig. 4: The path velocity profile of the robot with different velocity type motion parameter. From top left to bottom right are constant, from low to high and high to low, constant with pause, from low to high and high to low with pause.

parameters are designed. Since the base is only parametrized by the speed, I only explained the kinematics and the trajectory generation of the arm in this subsection.

1) Robot Platform: The robot used in this project is Omni-directional Assistive Robot, or Oarbot showed in Fig.1. The platform includes a 3 DoF mobile base which can moved in x-y direction and rotating in z-axis and a 6 DoF Kinova Curved-Wrist Arm. Please note that the arm is not a spherical arm which introduced harder inverse kinematics (IK).

2) Robot Kinematics: To drive the robot arm to the pose on the path, we have to know the joint angles through IK. There are two ways of doing this. One is to apply optimization through quadratic programming and approached the optimal angle that minimized the error between the target end-effector pose and the actual pose. Another way is by assuming the arm has a spherical wrist and solve it through subproblem decomposition. Although the second method introduced error in position, the orientation in the end-effector does match the target orientation. Since it is easier and more time efficient to perform IK through subproblem decomposition and I assume the little deviation in the position does not affect perceived emotion, I used the second method for the kinematics. Another possible way to do IK is by combining both. The arm first approach the optimal solution by solving IK with subproblem decomposition and then minimized the error with quadratic programming. The method not only increase the efficiency but also decrease the possibility that the quadratic programming stuck in a local minimal solution.

3) Trajectory Generation: To generate a trajectory based on the chosen set of motion parameters, a basic linear path is first generated passing through the start position and the ending position. The basic path is then average choosing N points which in this project $N = 50$ to explore with the exploration rate parameter (Fig.3). A new potentially wiggled path is then created. The waypoint is then sampled on the path the sampling rate in coordinate with the chosen arm speed which means that the time difference between each waypoint are the same. Finally, I adjust the position of waypoint on the path based on the velocity profile to reflect the velocity along the trajectory.

C. Motion Dataset Generation

As we are prepared with the motion synthesizing, I created 1000 motions by sampling the parameters in their designated range. The range of the parameters are as followed.

1) **Starting/Ending Position** r**:** 0.3 to 0.65 meter.
2) **Starting/Ending Position** θ**:** -60 to 90 degree.
3) **Starting/Ending Pitch:** -90 to 90 degree.
4) **Starting/Ending Base Height:** 0 to 0.3 meter.
5) **Starting/Ending Finger Position:** 0% to 100%.
6) **Exploration Rate** 0.01 to 0.1 meter.
7) **Arm Speed** 0.01 to 0.1 meter/second.
8) **Base Speed** 0.01 to 0.1 meter/second.
9) **Velocity Type** $6 kinds$.

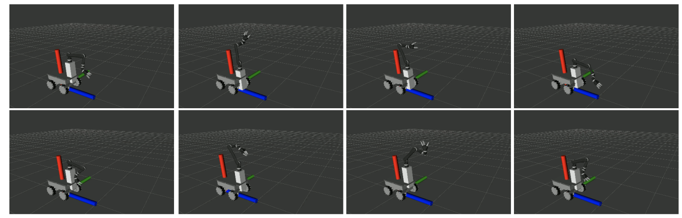

Fig. 5: 8 out of 1000 motions in the sampled motion set.

All the parameters are sampled uniformly in the range except that the exploration rate, arm speed and base speed are uniformly sampled in log scale to create more diverge motions. Fig.5 showed 8 out of 1000 motions in the motion set.

D. Evaluation of Perceived Motion

At this stage, we still don't know how the motions perceived or in other word, the emotion score of each motions. While we can label each motion with a emotion score, it is not practical and accurate to do so. Instead, I labeled the emotion score by first set up a tournament and comparing the happiness between each motion. The tournament is divided into three rounds. In the first and second round, each motions are compared with 5 other motions and 10 other motion in the last round. After each round, only the top half motion on the ranking will continue the tournament in the next round which means that only 500 and 250 motions were compared in the second and the third round. The structure not only increase the sampling efficiency by decreasing the comparison amount, it allows more comparison between more expressive motion than other ambiguous motions. The total amount of comparison with such structure is 5000.

Along the tournament, TrueSkill system is used to quantified the emotion score of each motions. The system model the score as a Gaussian distribution $G(\mu, \sigma)$ and updated the score after a comparison (a game). The parameter μ is associated to how many winning a motion has. and the parameter σ is associated to how many comparison a motion has been involved. I then finally used the μ value of each motion as its emotion score.

E. Mapping Motion Parameters to Emotions

The next step is to find the function that maps the motion parameters to emotions. I used the simple Linear Regression (LR) for to find the linear mapping. The emotion prediction function is then modeled as followed. β and β_0 are the coefficient and intercept. x is a vector of sampled motion parameters.

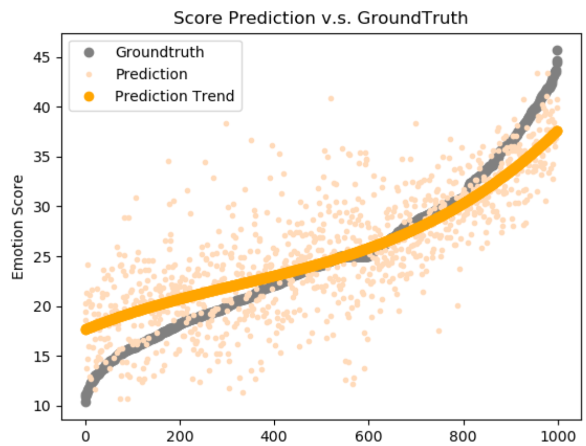

Fig. 6: The prediction value and the trend v.s. ground truth. The predicted value from the trained emotion prediction function followed the emotion score trend.

$$f(x) = \beta * x + \beta_0 \tag{1}$$

Before we applied LR, since the range of the parameters varies a lot, I applied standardization to the motion parameters. In particular, I applied the following equation.

$$\tilde{a} = \frac{a-\mu}{\sigma} \tag{2}$$

a is a specific motion parameter in the set. μ and σ are the mean and standard variance of all the sampled motion parameters of the same categories. To this points, we can applied LR. The question is then formulated as followed.

$$\begin{bmatrix} A & 1 \end{bmatrix} * \begin{bmatrix} \beta \\ \beta_0 \end{bmatrix} = b \tag{3}$$

In the LR problem, we want to know the coefficient vector β. The matrix A is the concatenated motion parameters of all motions in the set and the vector b is the corresponding emotion score vector. The coefficient vector β can be easily found by performing psuedo-inverse to the matrix A. I used Scikit-Learn library in Python for the project.

During the training stage, 80% of the data were used to trained the coefficient and the other 20% were used as test data. The predicted test data has the mean square error (MSE) 20.3141 and the R^2 score 0.6237. The R^2 score is relatively reliable and show consistent comparing to [1]. I further plotted the predict emotion score and the actual emotion score in Fig.6. We can see that the prediction value from the trained emotion prediction function followed the emotion trend.

F. Expressive Motion Generation

Finally we would like to generate a set of motion parameters given the emotion prediction function $f(x)$ and a emotion score. Because it's a multiple unknown and one equation problem, there are infinite solution. Although applying psuedo-inverse can find exact one solution, the solution is the one with minimal 2-norm and it is not necessary to do that. Therefore, I first randomly sampled one set of motion parameters, like what we have done in the data generation step but only one instead of 1000 sets. Then I minimized the error between the designated emotion score and the actual emotion score via gradient descent. Since the emotion prediction function is a linear map, the minimization can be achieved by the following equation.

$$x_d = \frac{y_d - \beta_0 - \beta^T x}{\beta^T \beta} \beta \tag{4}$$

x_d is the desired motion parameter sets. y_d is the desired emotion score. β and β_0 are the coefficient we learn in LR. Now we can finally generated motion with designated emotion score.

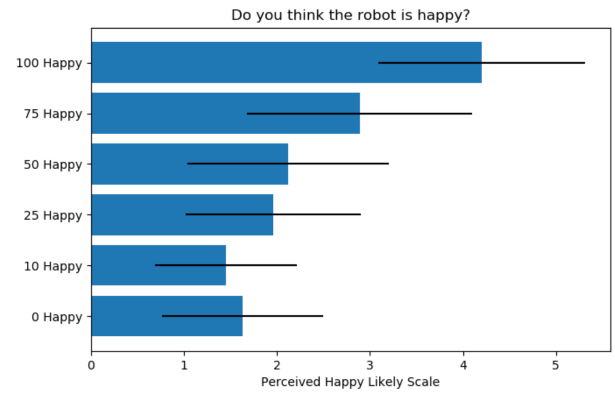

Fig. 7: Quantitative Evaluation Result. I used a questionnaire with 5-pt Likert-scale (anchors: 1 = not at all, 5 = strongly agree) to know how people perceived the motions. As the happy score getting higher, people do think the robot is happier.

IV. Evaluation

A. Quantitative Evaluation

Since we have the method to generate happy robot, I would like to know how well the model perform. To quantitatively evaluate the method, I leverage crowdsourcing.

First, 6 motions with $0, 10, 25, 50, 75, 100$ happy score are generated. 0 and 100 happy score are corresponds to the possible lowest and highest happy score that does not exceed physical motion limitation. 10 and 50 are the high and low bound in the training dataset and 25 is the middle points in the dataset. A questionnaire is created to ask "Do you think the robot is happy" with a 5-pt Likert-scale (anchors: 1 = not at all, 5 = strongly agree). Although in the previous section I mentioned it is not practical to give a score to happiness, it is much more easy when answering with a 5-pt Likert-scale. Moreover, it is possible to evaluate the motions using the tournament like how we label the dataset, but an actual good emotion expressive motion has to make people perceived the emotion even without comparison. Therefore, I chose to evaluate the motions with absolute scale.

There are total 73 responses for the evaluation. The result shows that the trained coefficient does know how to designed a happy motion. Fig.7 tells that as the happy score getting higher, people do think the robot is happier.

Fig. 8: Snapshot of real robot deployment. Using the pre-generated happy (Left) and unhappy (Right) motion joint trajectory, the robot is program to followed the joint trajectory with simple P-controller.

B. Real Robot Deployment

Finally, I deployed the generated happy and not so happy motion to the real robot. The robot is fed with a joint trajectory already generated and programmed to followed the joint with velocity input. A simple P-controller is used for joints trajectory following. Due to time limitation, the motions were only deployed on the arm of the robot.

V. Discussion

A. Goal Achieved

The project result successfully trained a LR model to generate happy motion for the robot and show the efficacy through quantitative evaluation. The project showed the framework is feasible and by a good design of motion parameters, the framework can be possibly generalized to other emotions or expressive behaviors. The project also deployed the motion to the real robot and proved it is the generated motions are physically possible to perform.

While the basic goal were achieved, the project only demonstrate the method on happy emotion due to the time limitation. In addition, only the arm part of the robot is used in the real robot deployment. Finally, the project does not include the human intention inference part as written in proposal. These are all good direction and can be a possible project for the future.

B. Happy Emotion and Others

While the evaluation showed that the method indeed generates happy motions, the original paper showed that the happy motions are among the easiest to learn. I conjecture that happy motion contains a lot of energetic actions which does not need specific action to expressive. On the other hand, emotion like surprised, disgust, required may required better motion parameter designed so that the motion can include other features.

C. Motion v.s. Pose

I included starting and ending pose as parameters because the previous literature suggest these can effect perceived motions. However, the goal of the project is to generate happy motions. Although happy "pose" may increase happiness in a motion, it may effect data labeling. For example, it is not a happy motion but the robot has a happy pose which is like a human wearing a happy pose but act very sad. A better way to label data is perhaps including neutral poses.

D. Assistive Arms v.s. Industrial Arms

In this project, I used Kinova arm ,which is an assistive arm, as the arm on the mobile platform. Although Kinova arm is a lot more safer to collaborate with, the velocity upper bound is very slow which restrict the diversity of the motion dataset. A lot of sampled motions act like a sloth or old people doing Tai-chi which is very difficult to tell the emotion from the motion.

VI. Conclusion

In this project, the goal is to have the robot learn to do emotional expressive motion. The overall framework followed [1] but used another kind of robot and designed new motion parameters. The projects first designed a few motion parameters based on what is observed in the previous literature. A motion dataset is then generated by sampling motion parameters and synthesizing motions. To give a suitable emotion score to each motion, instead of directly label the motion, a tournament is constructed and quantify the emotion score through TrueSkill library. After labeling the data, the emotion prediction function is learned from linear regression. The emotional expressive motion is finally generated by first random sampled a set of motion parameters and then achieved the dedicated emotion score in the direction of gradient. The generated happy motions were then evaluated by crowdsourcing to quantitatively evaluate the method. The result shows that people do feel the robot is happier when the emotion score is higher. The expressive motion is also deployed to the real robot to demonstrate the feasibility of these motions. Finally, a discussion was made on the goal achieved in this project, the happy emotion, motion and pose, and types of arms. I suggested that there should be more neutral pose during the data labeling stage and it is better to use arms with faster speed to increase the diversity of generated motion dataset. These are all good future research direction or project topics.

References

[1] R. Desai, F. Anderson, J. Matejka, S. Coros, J. McCann, G. Fitzmaurice, and T. Grossman, "Geppetto: Enabling semantic design of expressive robot behaviors," 04 2019, pp. 1–14.

[2] . J. O. Thomas, F., *Disney animation: The illusion of life*. New York: Abbeville Press, 1981.

[3] S. Hoshyari, H. Xu, E. Knoop, S. Coros, and M. Bächer, "Vibration-minimizing motion retargeting for robotic characters," *ACM Trans. Graph.*, vol. 38, no. 4, jul 2019. [Online]. Available: https://doi.org/10.1145/3306346.3323034

[4] M. K. Pan, S. Choi, J. Kennedy, K. McIntosh, D. C. Zamora, G. Niemeyer, J. Kim, A. Wieland, and D. Christensen, "Realistic and interactive robot gaze," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 11 072–11 078.

[5] Disney. Stuntronics. [Online]. Available: https://sites.disney.com/waltdisneyimagineering/stuntronics/

[6] B. Barnes, "Are you ready for sentient disney robots?" *New York Times*. [Online]. Available: https://www.nytimes.com/2021/08/19/business/media/disney-parks-robots.html

[7] H. Yi. Sozo artists. [Online]. Available: https://www.sozoartists.com/kuka

[8] T. Minka, R. Cleven, and Y. Zaykov, “Trueskill 2: An improved bayesian skill rating system,” Microsoft, Tech. Rep., March 2018. [Online]. Available: https://www.microsoft.com/en-us/research/publication/trueskill-2-improved-bayesian-skill-rating-system/

[9] J. Harris and E. Sharlin, “Exploring the affect of abstract motion in social human-robot interaction,” in *2011 RO-MAN*, 2011, pp. 441–448.

[10] H. Knight and R. Simmons, “Laban head-motions convey robot state: A call for robot body language,” in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 2881–2888.

[11] A. Zhou, D. Hadfield-Menell, A. Nagabandi, and A. D. Dragan, “Expressive robot motion timing,” in *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*, ser. HRI ’17. New York, NY, USA: Association for Computing Machinery, 2017, p. 22–31. [Online]. Available: https://doi.org/10.1145/2909824.3020221

[12] G. Venture and D. Kulić, “Robot expressive motions: A survey of generation and evaluation methods,” *J. Hum.-Robot Interact.*, vol. 8, no. 4, Nov. 2019. [Online]. Available: https://doi.org/10.1145/3344286