Analysis of Weight-Expressive Motion

16-741 Fall 2016

Rosario Scalise {rscalise}, Tanmay Batra {tbatra}, Sandeep Konam {skonam}

December 13, 2016

1 Introduction

When humans collaborate to jointly perform a complex manipulation task, what results is often a delicate dance brimming with subtleties. People take cues about their collaborator's intents based upon his or her motion and perform their own motions in an anticipatory manner so that their collective execution is as seamless and fluid as possible. In robotics we are concerned with emulating these kinds of natural human interactions such that one day humans can collaborate with robots in an equally effortless manner. In the past, robotics literature has focused on how to generate what is called legible motion [1] - motion that is intent-expressive. Recently, there has been work on extending this idea to use motion to not only convey intent about the goal, but also task context. In particular, many tasks rely on having a sense of the weights of the objects being manipulated. Since robots do not necessarily have the same range of weight-lifting capacities as humans and do not show when they are near their limits, human collaborators might have difficulties inferring the weights of objects their collaborative manipulator is handling. This could be especially problematic in the setting of object handovers where the robot is handing an object to a human. Generating weight-expressive motion solves this problem by allowing the robot to move such that its motion conveys how heavy the object being manipulated is.

In this project our goal is to explore the qualities that embody effective weight-expressive motion. Our approach is to perform two sets of analyses with a focus on the mechanics of the motions. We use a previously developed formalism for generating weight-expressive motion [2] on a two link planar robot (only two joints are active) holding objects of varying weights, and existing data from user feedback with respect to these motions. We consider the joint parameters values along waypoints of trajectories generated in differing classes of object weights and find the most discriminitive features among them. Additionally we consider the natural language data users give as reasoning for why they label a motion belonging to a 'heavy object' or a 'light object' manipulation task.

2 Generating Weight-Expressive Motion

2.1 Weight Inference

The previous work makes the assumption that an observer (human-user) has an estimate of the upper and lower torque bounds of the robot. Given this information and a robot trajectory, an optimal observer can then infer lower and upper bounds on the weight of the manipulated object, as shown below:

$$\begin{aligned} \underset{\tau}{\text{minimize}} \quad & f \\ \text{subject to} \quad & J^T(q)F = \tau - M(q)\ddot{q} - C(q,\dot{q}) - N(q,\dot{q}) \\ & \tau^l \le \tau \le \tau^u \\ \text{where} \quad & F = [0,0,1,0,0,0]^T f \end{aligned} \tag{1}$$

An illustration of this idea is shown in Fig. 1

Equation 1, defines $q \in \mathbb{R}^n$, $\dot{q} \in \mathbb{R}^n$ and $\ddot{q} \in \mathbb{R}^n$ the robot configuration, velocity and acceleration, where n the robot degrees of freedom; J is the Jacobian of the center of mass of the grasped object, assuming that it is rigidly attached to the end-effector frame. M is the manipulator mass matrix. C represents the Coriolis forces and N represents the gravitational and frictional forces. For given q, $\dot{q}$ and $\ddot{q}$, the constraints are linear, and the optimization can be solved efficiently using linear programming. The linear program above gives a lower bound on the weight of the end-effector f^l. By replacing the minimization with a maximization, we can get an upper bound f^u as well.

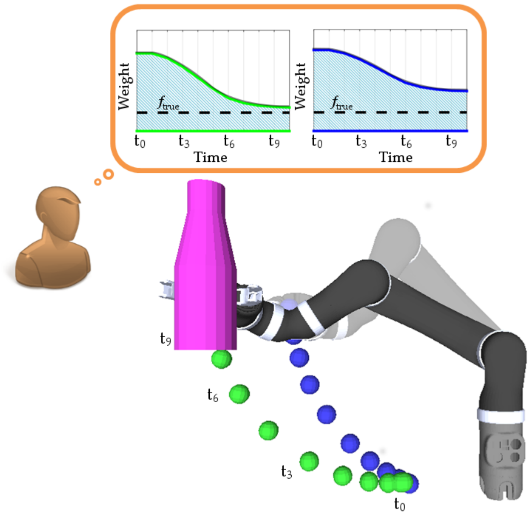

Figure 1: The green trajectory shows the optimized weight-expressive motion while the blue trajectory shows the unoptimized baseline motion. The user infers a better estimate of the weight bounds of the object for the optimized motion as indicated on the corresponding green curve.

2.2 Evaluating Weight-Expressiveness

A robot trajectory is parameterized by θ as a function of class C^2, so that $\xi(\theta, t) : \mathbb{R}^k \times \mathbb{R} \to \mathbb{R}^n$, where k equals the number of parameters. The assumption that the first and second derivatives with respect to θ and t are known is made.

To measure weight-expressiveness, previous work defines a cost functional that takes as its input a trajectory of duration T sampled at N_s equidistant time-steps, where $q(\theta, t_i)$, $\dot{q}(\theta, t_i)$ and $\ddot{q}(\theta, t_i)$ a sampled configuration, velocity and acceleration at time-step t_i, when $t_i = \{1\frac{T}{N_s}, 2\frac{T}{N_s}, ..., N_s\frac{T}{N_s}\}$. The functional returns a scalar cost $W : \Xi \to \mathbb{R}_+$. Given the model for weight inference described in Sec. 2.1, a weight-expressive trajectory minimizes the distance between the inferred f^l and f^u.

$$W[\xi] = \frac{\sum_{t_i} (f^u(\theta, t_i) - f^l(\theta, t_i))^2 w(t_i)}{\sum_{t_i} w(t_i)} \tag{2}$$

In Eq. 2, the function $w(t)$ is a weight function which allows for different parts of the trajectory to be weighted. We leave this in its default form for the purpose of the analysis in this project.

2.3 Inverting the Model to Generate Motion

We can now generate weight-expressive trajectories by minimizing the cost W given by Equation 2. In our case, $f^l(\theta, t_i)$ and $f^u(\theta, t_i)$ in Eq. 2 are not given by analytic expressions, but are computed by solving the linear optimization program of Eq. 1. Therefore, previous work formulates the problem as a non-linear optimization program and use a derivative-free solver to compute a solution in which we minimize the cost W defined in Eq. 2

$$
\begin{aligned}
\underset{\theta,T}{\text{minimize}} \quad & W[\xi] \\
\text{subject to} \quad & J^T(q(\theta,t_i))F_{true} - \tau(\theta,t_i) = M(q(\theta,t_i))\ddot{q}(\theta,t_i) \\
& \quad + C(q(\theta,t_i),\dot{q}(\theta,t_i)) + N(q(\theta,t_i),\dot{q}(\theta,t_i)) \\
& \tau_{MIN} \leq \tau(\theta,t_i) \leq \tau_{MAX} \\
& q^l \leq q(\theta,t_i) \leq q^u \\
& \dot{q}^l \leq \dot{q}(\theta,t_i) \leq \dot{q}^u \\
& T > 0 \\
\text{where} \quad & F_{true} = [0,0,1,0,0,0]f_{true} \\
\text{and} \quad & t_i = \{1\frac{T}{N_s},...,N_s\frac{T}{N_s}\}
\end{aligned}
\tag{3}
$$

The optimization constraints include the robot dynamics and the angle, velocity and torque limits of each joint. f_{true} is the true weight of the object, and q^l, q^u, $\dot{q}^l$, $\dot{q}^u$ are the joint angle and velocity limits.

2.4 Generating Trajectories

We generated 10 trajectories per object weight class (baseline, light {9N,....,19N} and heavy {29N,....,39N}) where each trajectory comes from an optimization for a different weight object. Each of the trajectories have 100 waypoints specifying the joint state configurations for j1 and j2 at each time step.

3 Collecting Labled Data from Users

A portion of the data analayzed in this project came from user-studies in which participants were shown videos of a simulated robot arm tracing out planar trajectories where the goal of the task is to lift a pink 'bottle-like' object above a certain height. (Please note that in the videos, we only controlled two joint values (j1,j2); the wrist joint values were passively set to keep the gripper holding the bottle level, and not included in the optimization). The final resting position of the end effector was underspecified in these simulations, and so the optimization would determine this depending on the weight of the object being lifted. The durations of each of the trajectories shown to the particpants were identical across classes.

The participants were shown trajectories of objects belonging to three classes. The first class was the unoptimized motion 'baseline' class, the second class was the optimized 'light object' class in which the object weight is specified to be 12 N, and the third class was the optimized 'heavy object' class in which the object weight is specified to be 32 N. Participants were told the pink container could be empty or contain a heavy substance, and then asked to label what kind of object the arm was lifting, light or heavy. They were also asked to answer a comparative example as shown in 2. Finally, for each of their answers they were asked to give their reasoning for their choice in a natural language free-response. Participants used language which could be easily mapped to properties of the mechanics of the motions (which was part of the motivation for this analysis project).

4 Analysis of User Feedback

The features of the motion include position, velocity and torque. Thus we look for words that have similar meanings to these features and find correlations of the frequency of these words with the labels assigned by the people. This analysis shows us about what people think when they feel if the object is heavy or light. Through this analysis we will find that if most of the important words in their explanations are related to the actual feature space (comprising of position, velocity and torque) or people think of some other completely different reasons to explain the weight of the object that have no relation with the feature of motion.

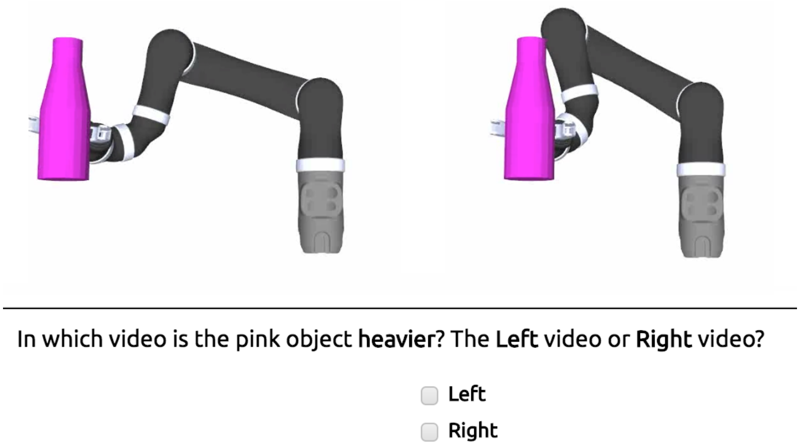

Figure 2: A screenshot of the user study asking the particpant to compare two trajectories.

Let's start the analysis for the videos where the object was heavy. We asked 100 people to label these videos and explain the reason behind their labels. We observed that out of 100, 93 people labelled it as heavy and 7 people labelled is as light. We can see that the level of accuracy for this particular case is very high and most of the people were easily able to distinguish the object in these videos as heavy. We also observed that the important words in their sentences are actually related to the features. We manually took out the important words from the sentences and clustered them together.

Firstly we'll show the clusters for the case when people correctly labelled the object as heavy. We found the following important clusters.

Words related to position: not able to raise high, raised closer to body, smaller range of motion

Words related to velocity: slow, longer time, a while to lift, less speed, steady, controlled.

Words related to torque: strain, effort, a lot of work, not able to lift, trouble in lifting, difficulty, struggle, work harder, tedious, wavering

The frequency of the velocity cluster was 79 i.e. in 79 out of 93 sentences people used the words present in that cluster. Similarly the frequency of the position cluster is 8 and the frequency of the torque cluster is 17. So we can see that people do come up with reasoning which is consistent with the feature space of the motion. The important words are mostly related to these features. An important observation here is that most of the reasoning is related to velocity i.e. people perceive that the object is heavy because the motion seems slow. This is the most important distinguishing factor for people to separate heavy and light object.

We would like to mention that since time is kept constant here and since velocity is directly proportional to position, even though people are not explicitly talking about position much, position is implicitly involved here as velocity and position are both interdependent in this case.

As we mentioned before, 7 times out of 93 people have incorrectly labelled heavy object as light one. The clusters we observed for this case are as follows:

Words related to torque: no acceleration (even though speed is slow), less force

Words related to position: raised high

We can see here that these are exactly opposite to the explanations we saw in the previous case. This means that even when people are labelling it wrong, they are still consistent with their reasoning. The reasoning is still very much related to the feature space. An additional thing we observed in this case is that some explanations are related to object priors that people may have. For example, people just have a prior that bottles are light and so they labelled it as light.

For the next case, we analyse the sentences for the videos where the object was light. We observed that out of 97 labelled videos, only 28 times they were labelled correctly as light and 69 times they were incorrectly labelled as heavy. This shows that when the object is light, it is very difficult for the people to actually label it as light.

The clusters for the case when people correctly labelled it as light are shown below:

Words related to position: lifted higher, away from body
Words related to velocity: fast, quick, smooth
Words related to torque: no issues in lifting, easy, less power, seamlessly, not too much effort

The clusters for the case when people incorrectly labeled it as heavy are shown below:

Words related to position: limited range of motion Words related to veolcity: slow, longer time, a while to lift, controlled Words related to torque: struggle, wobbly, tilting, difficult, more force, hesitation, wavering, straining, more effort

An interesting thing to observe here is that the explanations are consistent with what we saw in the previous case when the ground truth was heavy. This it seems that people are consistent in their thinking irrespective of what the ground truth is. This is an important part of this study that people actually consider words related to the feature space as distinguishing factor between the two categories.

5 Analysis of Mechanics of Motion

5.1 Information about data

Through our experiments, we collected data corresponding to three classes : heavy, light and baseline. Each class has 10 trajectories, where each trajectory corresponds to a different weight. Each trajectory has data corresponding to 100 timestamps. Each time-stamp has values for 6 parameters - Position, Velocity, and Torque respectively at the two joints. For the purpose of analysis, we split the dataset of 10 trajectories into 7 and 3 trajectories for training and testing respectively.

5.2 Data Preparation

Taking advantage of the time-series data, we wanted to analyze how the 6 parameters varied across the waypoints and understand how it varied across the three classes. Taking a naive approach, we concatenated the 6 parameters corresponding to all the waypoints. Therefore, for training data, we had 21 rows, where every 7 rows belonged to one of the three classes, and 600 columns, where every 6 columns corresponded to the 6 parameters of one trajectory. At the end of the data preparation step, we had a matrix of 21 rows and 600 columns corresponding to the training data.

5.3 Algorithms

For the purposes of classification, we used L1-regularized SVM classifier. The support vector machine (SVM) is a widely used tool for classification. It was first motivated by the geometric consideration of maximizing the margin. If given a set of training data (x_1, y_1), . . . ,(x_n, y_n), where the input $x_i \in R^p$ is a vector with p predictor variables, and the output $y_i \in (1, -1)$ denotes the class label, the SVM finds a hyperplane that separates the two classes of data points by the largest distance:

$$\max_{\beta,\beta_0} \frac{1}{\|\beta\|_2{}^2}; y_i(\beta_0 + {x_i}^T\beta) \geq 1 - \xi_i, \xi_i \geq 0, \sum_{i=1}^{n} \xi \leq B$$

where ξ_i are slack variables that describe the overlap between the two classes, and B is a tuning parameter that controls the overlap.

L1-regularized classifiers can be used to obtain a sparse model. Given a set of instance-label pairs (x_i, y_i), i = 1, . . . , l, $x_i \in R^n$, $y_i \in (-1, +1)$, training an L1-regularized linear classifier involves the following unconstrained optimization problem:

$$\min_{w} f(w) \equiv \|w\|_1 + C \sum_{i=1}^{l} \xi(w; x_i, y_i)$$

where $\|.\|$ denotes the 1-norm and $\sum_{i=1}^{l} \xi(w; x_i, y_i)$ is a non-negative (convex) loss function. The regularization term $\|w\|_1$ is used to avoid overfitting the training data. The user-defined parameter $C > 0$ is used to balance the regularization and loss terms.

5.4 Experiments and Results

We ran three different sets of experiments, each one aimed at better understanding the data.

5.4.1 Case-1

During the first experiment, we uniformly sampled the waypoints and selected the parameters corresponding to 1st, 11th, ...99th waypoints. Training data matrix in this case is 21 rows and 60 columns. After passing through the classifier, we analysed the weights corresponding to all the values in the training data. We observed an interesting pattern that, only values corresponding to positions of the second joint are non-zero. Intuitively, it makes sense that position of the second joint is important because it ends up having most prominent motion out of the two joints. Position is more important than velocity because it varies more. In the position, second one makes more contribution to the end effector.

5.4.2 Case-2

During the second experiment, we considered all the waypoints and the corresponding parameters. Training data matrix, in this case, is 21 rows and 600 columns. After passing through the classifier, we analyzed the weights corresponding to all the values in the training data. We observed the same pattern as in the Case-1, i.e. only position of the second joint has a non-zero value.

5.4.3 Case-3

During the third experiment, we analysed each of the 6 parameters separately. Therefore, each of the matrices had 100 columns corresponding to the value of each parameter at that particular waypoint. In this case, there are 6 different training data matrices, each one with 21 rows and 100 columns. After passing through the classifier, we analysed the weights corresponding to all the values in the training data. Since the training data had 6 distinct matrices, we had 6 distinct weight matrices. In the weight matrices corresponding to position, we observed that majority non-zero values occurred after the timestamp of 80. Since in all the cases, manipulator starts with the same position, but ends up at different positions, there is higher variance towards the end. In the weight matrices corresponding to velocity, we observed that non-zero values occurred between timestamps of 40 and 90. Since velocity varies with more magnitude initially and towards the end, there is more stability in the mid-range. In the weight matrices, corresponding to torque, we observed that non-zero values occurred between timestamps of 20 and 70.

6 Conclusion

In line with our hypotheses, we found that the most significant determiner of successful weight-expressive motion are the joint velocities, particularly throughout the latter half of the trajectory. This was supported by the analysis of participant reasoning when succesfully labeling trajectories. Interestingly, our algorithmic analysis indicated that joint angle position was the biggest discriminating feature (in particular the second joint angle position). We account for this by making note that people implicitly reason about joint angle position when citing velocity as their primary discriminators since the duration of each trajectory is held constant.

7 Limitations and Future Work

In general, it is difficult to say with any certainty to what extent implicit reasoning about position through velocity plays a role in successful object weight inference. In order to yield more conclusive results, we could deploy a follow-up user study in which we remove possible confounds. For example, our samples are not all i.i.d. which weakens the efficacy of the modeling techniques. Additionally, analyzing the end effector position in cartesian space rather than the joint angle position values could provide more insight, however it important to note this is note explicitly used in the optimization procedure and is only a side-effect.

References

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

[2] S. Nikolaidis, R. Scalise, and S. S. Srinivasa, "A formalism for generating weight-expressive motion," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on (in review)*. IEEE.