CS 199 Quarter Report: Visualizing Human Utility from Video Demonstrations for Deductive Planning in Robotics

Kang (Frank) Chen
University of California, Los Angeles

Abstract

This work, in direct support of the IJCAI paper on learning human utility from video demonstrations for deductive planning in robotics, involves generating utility landscapes that define the desired states of our shirt. We performed cross validation to fine-tune our hyperparameters for the learning algorithm support vector regression (SVR). Furthermore, we collected human preferences on various scenarios of shirt folding in order to demonstrate that the learned preference model strongly matches preferences of 335 human decisions, capturing the commonsense goal.

1 Introduction

As a continuation from Fall Quarter, we specifically want to teach a robot how to fold shirts through human demonstrations, and have it reproduce the skill under both different articles of clothing and different sets of available actions. Our experimental results show good performance on a two-armed industrial robot following causal chains that maximize a learned latent utility function. Most importantly, the robots decisions are interpretable, facilitating immediate natural language description of plans. Human preferences are modeled by a latent utility function over the states of the world. To rank preferences, we pursue relevant fluents of a task, and then learn a utility function based on these fluents. For example, Figure 4 shows the utility landscape for a cloth-folding task, obtained through 45 visual demonstrations.

Figure 1 shows how fluents in the shirt are captured; we implemented this fluent extraction code in Winter 2016, as part of our work in fluent extraction sent to NIPS 2016. In order to produce a visualizatin, we use MDS to reduce the dimensionality to a 3 dimensional space.

The ranking formulations detailed in [13] are beyond the scope of this report, as the goal of this report is to explain how we accomplished the following tasks:

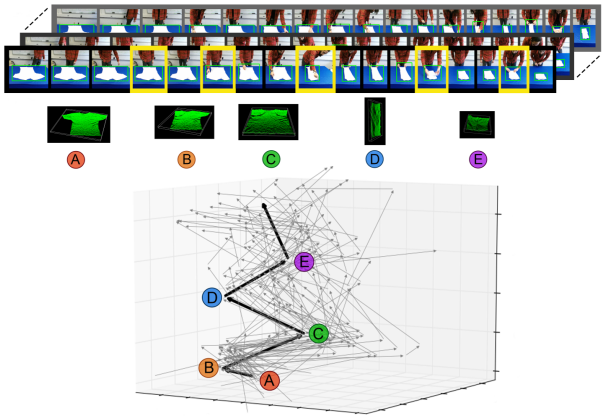

Figure 1: In the process of cloth-folding, the 12 fluents of a shirt drift through a 12-dimensional space. The diagram visualizes a human demonstration as a trajectory of fluents (using MDS to reduce dimensionality to 3 dimensions).

1. Performing cross validation to fine-tune SVR hyperparameters
2. Visualizing utility landscape of shirt-folding and its surrounding world
3. Surveying and visualizing human preferences

2 Related Works

Our work relates most closely to the following three categories.

Modeling and Learning Human Utility: Building computational models for human utilities could be traced back to the English philosopher, Jeremy Bentham, in his works on ethics known as utilitarianism [3]. Utilities, or values, are also used in planning schemes like Markov decision process (MDP) [8], and are often associated with states of a task. However, in the literature of

planning, this type of "value" is not a reflection of true human preference and, inconveniently, is tightly dependent on the agent's actions.

Zhu *et al*. [12] first modeled human utilities over physical forces on tools, and proposed effective algorithms to learn utilities from videos. In this work we also adopt ranking theories, but automatically pursue relevant fluents based on human demonstrations. We integrate their ideas to drive goal-driven robot behavior.

Inverse Reinforcement Learning: Inverse reinforcement learning (IRL) aims to determine the reward function being locally optimized from observed behaviors of the actors [1]. In IRL, the observed behaviors are often assumed to be optimal. Hadfieldmenell *et al*. [5] defined cooperative inverse reinforcement learning (CIRL), which allows reward learning when the observed behaviors could be sub-optimal, based on human-robot interactions. In contrast to IRL or CIRL, our method does not assume any kind of optimality, nor is our approach dependent on the set of possible actions. It avoids the correspondence problem in human-robot knowledge transfer by learning the global utility function over observed states, rather than learning the local reward function from actions directly.

Robot Learning from Demonstrations: Learning how to perform a task from human demonstrations has been a challenging problem for artificial intelligence and robotics, and various methods were developed trying to solve the problem [2]. Learning the task of cloth-folding from human demonstrations, in particular, has been studied before by Xiong *et al*. [11]. While most of the existing approaches focus on reproducing the demonstrator's action sequence, our work tries to model human utilities from observations, and generates task plans deductively from utilities.

3 Experiment Formulation

3.1 Fine-tune SVR hyperparameters

The SVR function in scikit-learn has many hyperparameters; the ones we are interested in are:

1. **kernel**: specifies the kernel type to be used in the algorithm. It must be one of 'linear', 'poly', 'rbf', 'sigmoid'. 'rbf' is used as default.

2. **C**: penalty parameter of the error term.

3. **gamma**: Kernel coefficient for 'rbf', 'poly', and 'sigmoid'.

4. **epsilon**: epsilon in the epsilon-SVR model. It specifies the epsilon-tube within which no penalty is associated in the training loss function with points predicted within a distance epsilon from the actual value.

5. **shrinking**: whether to use the shrinking heuristic. True/False

We want to determine which combination of hyperparameters produce the best results. We used the mean squared error as our loss function to cross validate all permutation of hyperparameters.

3.2 Visualize utility landscape

We want to create a 3-dimensional visualization of our utility landscape, where the z-axis indicates the increasing level of utility of a state. For this implementation, we first pre-processed our data into 3 dimensions using MDS, then performed k-means clustering to find centroids of our data; these centroids become the various 'states' our shirt is in during the process of cloth-folding.

We learn our utility function from 45 RGB-D (*pointcloud*) video demonstrations of t-shirt folding. This dataset splits into 30 for training and 15 for testing. The videos were recorded on a separate Kinect camera at a different orientation in a separate setting by different people.

At each frame, the vision processing step segments the cloth using graph-based techniques on the 2D RGB image [4, 9], and then the extracted 3D cloth pointcloud is aligned to its principal axis. Next, we extract fluents from the pointcloud being tracked. Examples of a couple fluents include width, height, thickness, x-symmetry, y-symmetry, and the 7 moment invariants [6].

We extract both automatic and hand-designed fluents to obtain a training dataset $\mathscr{X}$ to pursue relevant fluents and obtain an optimal ranking as outlined in Algorithm 1. Each utility function is approximated by a piecewise linear function. The vision processing code, ranking pursuit algorithm, and the RGB-D dataset of cloth-folding are open-sourced on the author's website [13].

The ranking formulations are beyond the scope of this report, and can be found in [13]. We have included a copy of the ranking pursuit algorithm in this paper for clarity.

Lastly, we added some noise to the data in order to obtain a landscape that contained lower utility contours for areas that are not part of the utility desires for shirt folding. We did so by generating random points that are far away euclidean distance-wise from the current centroids. These points represent the rest of the world, and as shown in Figure 4, are at the lowest z-axis utility.

Algorithm 1 Ranking Pursuit

procedure UTILITYESTIMATION(F)
  Initialize $\lambda_0^{(\alpha)}$ for each $f^{(\alpha)} \in F$
  while KKT conditions not met **do**
    $\lambda_{t+1}^{(1)} \leftarrow \text{Update}(\lambda_{t+1}^{(1)} | \lambda_t^{(2)}, ..., \lambda_t^{(k)}, \mathscr{X})$
    $\lambda_{t+1}^{(2)} \leftarrow \text{Update}(\lambda_{t+1}^{(2)} | \lambda_{t+1}^{(1)}, ..., \lambda_t^{(k)}, \mathscr{X})$
    ...
    $\lambda_{t+1}^{(k)} \leftarrow \text{Update}(\lambda_{t+1}^{(k)} | \lambda_{t+1}^{(1)}, ..., \lambda_{t+1}^{(k-1)}, \mathscr{X})$
  return $\lambda^{(1)}, ..., \lambda^{(k)}$
procedure FLUENTSELECTION(F)
  $minError \leftarrow \infty$
  for $f^{(\alpha)} \in \Omega_F$ **do**
    $\hat{F} \leftarrow F \cup \{f^{(\alpha)}\}$
    $\Lambda \leftarrow \text{UtilityEstimation}(\hat{F})$
    if $\text{Violations}(\Lambda, \mathscr{X}) < minError$ **then**
      $minError \leftarrow \text{Violations}(\Lambda, \mathscr{X})$
      $F^* \leftarrow \hat{F}$
  return F^*

3.3 Analyzing human preferences

We perform external evaluations on 335 individuals to compare how well the learned preference model matches human judgement. In this survey, each human was asked to make a decision on 7 choices after being told, "A robot attempts to fold your clothes. Of each outcome, which do you prefer?"

Figure 2 shows a sample question that we asked on the online poll. After gathering around 335 human responses, we parsed the data and encoded it as follows:

$$\text{Option A} \rightarrow 1$$
$$\text{Option B} \rightarrow 0$$

From this encoding, we were able to generate a 1×7 "preference vector" that indicates the series of preferences for each user. From there, we generated a histogram of our preference vectors in order to visualize which set of preferences occured most frequently in our sample of 335 participants.

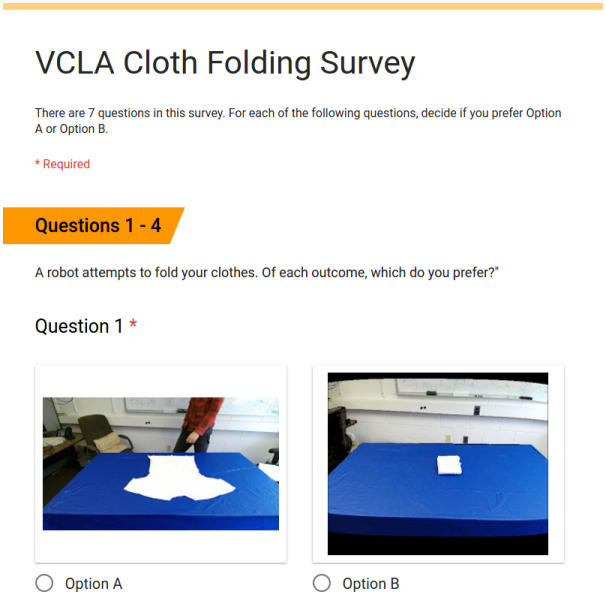

Figure 2: A sample question from the online poll. Each question has two options, A and B; participants will choose one of the two options that they prefer.

4 Results

Figure 3 shows our initial efforts in generating a utility landscape. The issue with this visualization is that the unfolded shirt lies in the valley of the utility function, but that is not an accurate interpretation, as other objects unrelated to the desired state (ex. a table, or a pair of pants) should have even less utility. We made adjustments to our landscape visualization by sampling random points and assigning lower utility values to them. After combining these "lower utility" points with the points that

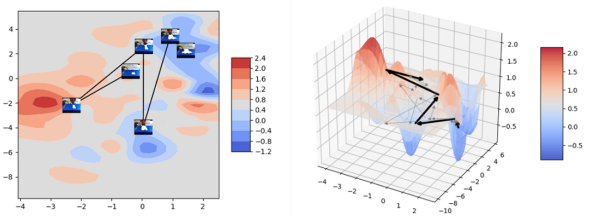

Figure 3: The utility landscape identifies desired states. This one, in particular, is trained from 45 cloth-folding video demonstrations. For visualization purposes, we reduce the the state-space to 2 dimensions through multi-dimensional scaling (MDS).

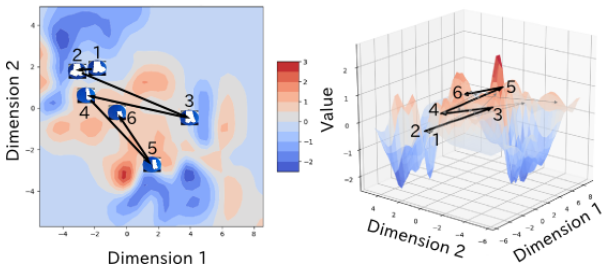

Figure 4: The utility landscape identifies desired states. This one, in particular, is trained from 45 cloth-folding video demonstrations. For visualization purposes, we reduce the the state-space to 2 dimensions through multidimensional scaling (MDS). The canyons in this landscape represent wrinkled clothes, whereas the peaks represent well-folded clothes. Given this learned utility function, a robot chooses from an available set of actions to craft a motion trajectory that maximizes its utility.

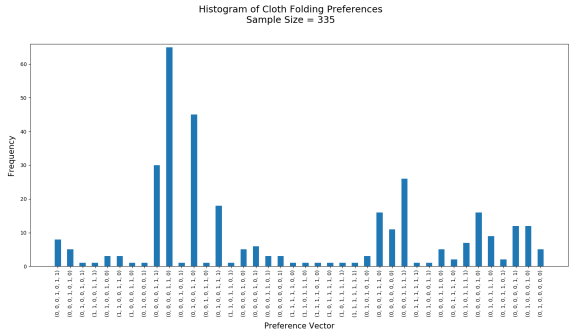

Figure 5: Histogram of cloth folding preferences.

correspond to our own utility landscape, we generate a visualization that more closely resembles a typical utility landscape in Figure 4.

The histogram of our human preference frequency is shown in Figure 5. There is a clear preference vector that has more weight than the rest of the preference combinations; we can use this knowledge to determine if our robot's actions closely resembles that of the human preference.

5 Future Works

In this work, we have implemented a deductive learning approach that demonstrates how a latent utility function can generalize robot task understanding. Specifically, the tasks outlined in this report include cross validating hyperparameters for SVR, visualizing utility landscape of shirt fluents, and learning preference models from human participatns.

As shown in [13], the learned preference model strongly matches preferences of 335 human decisions, capturing the commonsense goal. The inferred action plan is not only interpretable (*why*, *how*, and *what*), but also performable on a robot platform to complete the learned task in a completely new situation.

In future work, we would like to incorporate social choice theory [10] and understand inconsistencies between human preferences [7]. Furthermore, we would like to resign this task dependent perspective of human demonstrations, and instead focus on abstractions of fluents to generalize knowledge across different types of tasks (cloth-folding, tea-making, desk-organizing, etc.) [13].

6 Acknowledgements

I would like to thank PhD student Nishant Shukla and Prof. Song-Chun Zhu for their mentorship and guidance.

References

[1] ANDREW Y. NG, S. R. Algorithms for inverse reinforcement learning. In *Proceedings of International Conference on Machine Learning (ICML 2000)* (Stanford, USA, June 2000).

[2] ARGALL, B. D., CHERNOVA, S., VELOSO, M., AND BROWNING, B. A survey of robot learning from demonstration. *Robotics and autonomous systems 57*, 5 (2009), 469–483.

[3] BENTHAM, J. *An Introduction to the Principles of Morals and Legislation*. 1789.

[4] FELZENSZWALB, P. F., AND HUTTENLOCHER, D. P. Efficient graph-based image segmentation. *Int. J. Comput. Vision 59*, 2 (Sept. 2004), 167–181.

[5] HADFIELDMENELL, D., DRAGAN, A., ABBEEL, P., AND RUSSELL, S. Cooperative inverse reinforcement learning.

[6] HU, M.-K. Visual pattern recognition by moment invariants. *IRE transactions on information theory 8*, 2 (1962), 179–187.

[7] JIANG, X., LIM, L.-H., YAO, Y., AND YE, Y. Statistical ranking and combinatorial hodge theory. *Mathematical Programming 127*, 1 (2011), 203–244.

[8] PUTERMAN, M. L. Markov decision process. *Journal of the Royal Statistical Society 158*, 3 (1994), 1–16.

[9] ROTHER, C., KOLMOGOROV, V., AND BLAKE, A. "grabcut": Interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph. 23*, 3 (Aug. 2004), 309–314.

[10] SEN, A. Social choice theory. *Handbook of mathematical economics 3* (1986), 1073–1181.

[11] XIONG, C., SHUKLA, N., XIONG, W., AND ZHU, S. Robot learning with a spatial, temporal, and causal and-or graph. In *Proceedings of International Conference on on Robotics and Automation (ICRA 2016)* (Stockholm, Sweden, May 2016).

[12] ZHU, Y., JIANG, C., ZHAO, Y., TERZOPOULOS, D., AND ZHU, S. C. Inferring forces and learning human utilities from videos. In *IEEE Conference on Computer Vision and Pattern Recognition* (2016), pp. 3823–3833.

[13] ZHU, NISHANT S., Y. H., AND C., K. F. Markov decision process. *International Joint Conference in Artifical Intelligence* (2017).