Pointing Estimation for Human-Robot Interaction Using Hand Pose, Verbal Cues, and Confidence Heuristics

Andrew Showers(✉) and Mei Si(✉)

Rensselaer Polytechnic Institute, Troy, NY 12180, USA
{showea,sim}@rpi.edu

Abstract. People utilize pointing directives frequently and effortlessly. Robots, therefore, will need to interpret these directives in order to understand the intention of the user. This is not a trivial task as the intended pointing direction rarely aligns with the ground truth pointing vector. Standard methods require head, arm, and hand pose estimation inhibiting more complex pointing gestures that can be found in human-human interactions. In this work, we aim to interpret these pointing directives by using the pose of the index finger in order to capture both simple and complex gestures. Furthermore, this method can act as a fall-back for when full-body pose information is not available. This paper demonstrates the ability of a robot to determine pointing direction using data collected from a Microsoft Kinect camera. The finger joints are detected in 3D-space and used in conjugation with verbal cues from the user to determine the point of interest (POI). In addition to this, confidence heuristics are provided to determine the quality of the source information, whether verbal or physical. We evaluated the performance of using these features with a support vector machine, decision tree, and a generalized model which does not rely on a learning algorithm.

Keywords: Pointing · Object detection · Object localization
Social interaction

1 Introduction

Pointing directives are commonly used in social interactions to express an object of interest. In order for social robots to have meaningful interactions with people, interactions like these will need to be implemented. Unfortunately, systematically interpreting this seemingly trivial behavior has proved to be a difficult task. The difficulty stems from the number of factors influencing the pointing vector. A common solution in minimizing this error is to detect the head pose [7,8] or the gaze direction [2]. However, this correction requires additional information which may not be available due to occlusion. Ideally the system would have a more reliable fall-back method than using the hand pose exclusively.

G. Meiselwitz (Ed.): SCSM 2018, LNCS 10914, pp. 403–412, 2018.
https://doi.org/10.1007/978-3-319-91485-5_31

In this work, we propose to integrate contextual information for helping decoding the users' intentions. In order to frame the pointing direction within a given context, we use natural language processing as well as confidence heuristics to determine the accuracy of the information sources. The verbal component can extend to a large range of descriptors, including object names, attributes, and spatial references. However, in order to use this information the system needs to be aware of the user's environment. Fortunately, with the advancement of object detection and localization, this task is becoming feasible. By using object detection to learn about the environment, we can now leverage verbal cues as a way to filter out irrelevant objects.

We claim that the proposed method which combines these channels of information can interpret pointing directives significantly better than with the hand pose alone. Separately, the hand pose is too unreliable as shown by previous studies and using language alone may lead to overly complex and unnatural descriptions. By fusing these channels together we can attain a reliable and natural way of handling this social interaction.

2 Related Work

Previous studies have attempted to use the vector from the head-hand line [9], head-finger line [10], and forearm direction [4] as the pointing direction. Unfortunately these models lack accuracy. An example of this inaccuracy can be seen in the study performed by Abidi et al. [2] where the vector formed from the elbow to the hand/finger provides unsatisfactory results when trying to guide a robot. They found that using the pointing vector as the only feature led to 38% satisfaction. Improvements were made by combining the pointing vector with the gaze direction. However this relies on additional features that may not be easy for a robot to obtain or susceptible to noise.

In order to address this, a study performed by Ueno et al. [5] created a calibration procedure where the user points to the camera directly before pointing to the intended object. Offsets are determined by the difference in the vector formed from the eyes to the fingertip and the vector from the eyes to the target object. This method achieved accuracy between 80%–90% depending on the camera position.

An alternative approach is to apply learning algorithms for finding a transformation from a set of observed spacial features to the intended pointing direction. Droeschel et al. [3] used Gaussian Process Regression (GPR) to dramatically reduce the error of the desired pointing direction angles in comparison to the source vector. Similarly, Shukla et al. [6] implemented a probabilistic model to learn the pointing direction by providing hand pose images. This approach can be extended to other subproblems as well. Pateraki et al. [7] implemented a least squares matching technique to minimize error with estimating the head pose.

3 Proposed Method

We propose a computational model that takes verbal and visual input. The verbal input has multiple sources of ambiguity which can be observed. The message could become corrupted due to a failure in the Automatic Speech Recognition (ASR), leading to a loss of keywords and potentially the intention of the user. Even when the message is accurately retrieved, the information provided can have varying levels of description. The level of detail required to resolve this ambiguity is too verbose in comparison to what is naturally practiced. Furthermore, identifying what information to search for is difficult given free expression. Similarly, visual input is susceptible to noise and may lack contextual information not captured by the hand pose. Since both inputs carry inherent ambiguity, we adjust the system's trust on them dynamically.

3.1 Preprocessing

Knowledge of the objects and their respective position within the room is required for resolving the pointing directive. This knowledge is acquired with the use of an additional camera at a known location in the room. The image captured is fed into a Single Shot MultiBox Detector (SSD) [1] for object detection and localization. The model is pre-trained on the VOC0712 dataset containing 20 classes. The localization only provides a bounding box on the input image which does not provide the required spatial information of the objects (Fig. 1).

Fig. 1. Layout of the room

While the object position can be inferred from the Kinect depth stream, we used AR markers as a simple alternative. Given the limited classes within the model, we manually encoded 3 AR tags as object type "book". To provide some attribute information to the objects, we specified the color of each object by hand (Fig. 2).

Fig. 2. Object detection and localization with a Single Shot MultiBox Detector

3.2 Capture Process

In order to begin the capture process, subjects must use explicit keywords such as "get" or "bring" in their pointing directive. Messages were captured using a microphone and converted to text using a natural language processing library. To estimate the pointing direction, we take the line formed by the base and tip joints of the index finger. This information was captured by the Kinect using the Metrilus Aiolos finger tracking library. Spatial instances were collected until a minimum sample size, in our case 10, was reached in order to minimize noise.

3.3 Feature Vector

Physical Information. The pointing direction, i.e. the line formed between the base and tip joints, is not explicitly added to the feature vector to minimize over-fitting. Instead, the single direction is abstracted to a range of possible directions using the 5 closest objects as features. This gives a general sense of the direction while being too general to attribute to a specific sample instance.

Verbal Information. From the original message, the following categories are extracted; object names, object attributes, and spacial cues. Specifically, the number of references to each object type (e.g. "book") as well as boolean values for the presence of spacial and attribute keywords. For this paper, we define spacial cues to be keywords which may provide spacial information of the desired object, such as "bottom" and "left".

Confidence Heuristics. A useful metric to provide is the accuracy of the information source. To accomplish this, confidence heuristics were implemented for both verbal input and visual inputs. For the verbal component, weights were

applied to the mention of known attributes in the workspace, the number of spacial cues, and the message length and structure. For the physical component, confidence was inversely related with the variance of the capture buffer.

3.4 Model

In order to examine the importance of these features we trained a Support Vector Machine (SVM) and Decision Tree (DT) to classify the POI. The SVM used a linear kernel and the DT was limited to a max depth of 8. By tuning these models and examining the accuracy we were able to construct a simpler and more generalized model using the features provided to the learning algorithms.

4 Experiments and Results

4.1 Experiment Setup

In order to evaluate the accuracy of our approach, we conducted the experiments in an indoor environment. There are 9 test objects in the room; 3 bottles, 3 books, and 3 chairs. The subject stands in a fixed location in the room, approximately 8–10 ft away from the objects. The object positions relative to the Kinect are shown in Fig. 3. In this configuration, objects are non-unique and similar objects are spatially near one-another (e.g. the stack of books).

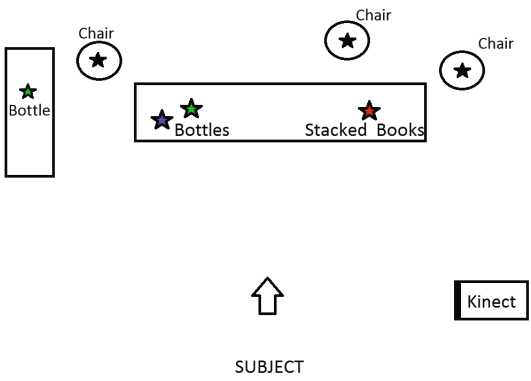

Fig. 3. Position of objects, Kinect, and the subject.

4.2 Procedure

For this experiment, we used 10 subjects, 7 male and 3 female. In order to determine the POI, the features provided are fed into the SVM and DT classifiers respectively. For the generalized model we first eliminate unrelated objects based on the verbal information provided, if any. Then by using this reduced set of

objects, the POI is either assigned to the object closest to the line formed by the pointing vector. In the event of low confidence on either of the input channels, the low confidence channel is discarded. We evaluated the accuracy of this method under the following conditions:

1. Pointing Independently – not allowed to talk
2. Subject can only express the object name and some attribute while pointing.
3. Subject can only express spacial information in relation to other objects while pointing.
4. Subject can freely express any information about the object while pointing.

For condition (4), subjects were required to point to 4 objects of their choosing. This relaxed condition was used to learn what subjects naturally say while identifying objects. For each of the other conditions, subjects were given a randomly generated permutation of the objects in which to point to. Given this setup, subjects were required to give 31 pointing gestures each, providing a total of 310 samples for the entire experiment.

4.3 Results

In order to evaluate the performance of the learning algorithms, the samples were divided into two datasets for training (70%) and testing (30%) respectively. Figures 4 and 5 show the accuracy of the SVM and DT on the test dataset. Notice Fig. 6 uses the entire dataset as there is no learning algorithm involved with the generalized approach. Additionally, each method is compared to using the pointing direction alone. As seen by Figs. 4, 5 and 6, this method is very inaccurate. On average, only 68% of the samples included the desired object when taking the 5 closest objects to the pointing vector. Without any learning

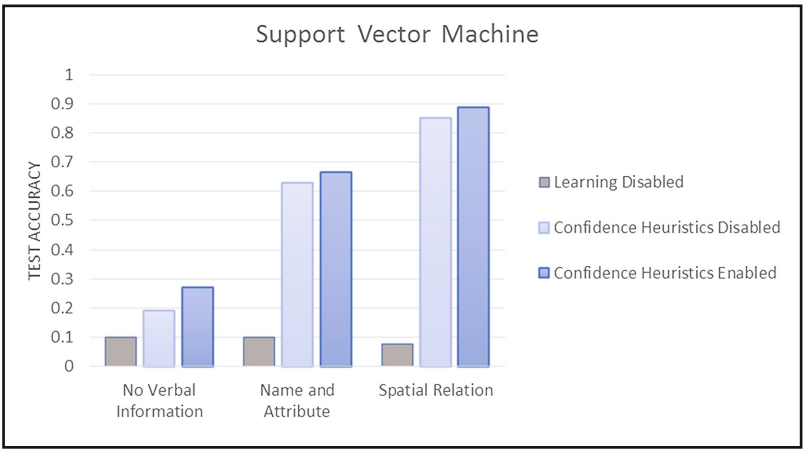

Fig. 4. Accuracy of the support vector machine on the test dataset

algorithm enabled the closest object is taken as the POI which leads to an accuracy rate no better than guessing.

The SVM was able to take advantage of additional information while managing to generalize well to new samples. In all cases the inclusion of confidence estimates for the data source led to improvements in performance. Even in the presence of no verbal information the accuracy significantly improved from 10% to 19% without heuristics. This improvement suggests patterns in the detected object positions vs their actual locations. Since the positions are detected using a combination of the SSD object localization and AR tags, these positions are susceptible to noise. Accuracy raised further to 27% with confidence heuristics enabled suggesting the existence of noise while estimating the pointing direction, such as potential dead-zones where the Kinect struggles to accurately detect the hand.

Introducing verbal information containing the object name and attribute significantly raised accuracy as would be expected. Accuracy improved to 63% and 67% with heuristics disabled and enabled respectively. Interestingly, complex verbal cues encoding spatial relations led to the highest performance in this experiment. Accuracy reached 85% with spatial cues and raised further to 89% with the inclusion of confidence metrics.

In comparison to the SVM, the DT was able to achieve similar performance for samples in which there was limited or no verbal information provided. With heuristics disabled, the DT was able to achieve accuracies of 12%, 67%, 77% for conditions (1), (2), and (3). Unfortunately decision trees are prone to overfitting, which can be seen by the decreased accuracy when introducing confidence heuristics. The tree incorporated the heuristics but the model poorly generalized to new samples. With heuristics enabled, the DT was able to achieve accuracies of 27%, 60%, 56% for conditions (1), (2), and (3). As seen in Fig. 5, as the

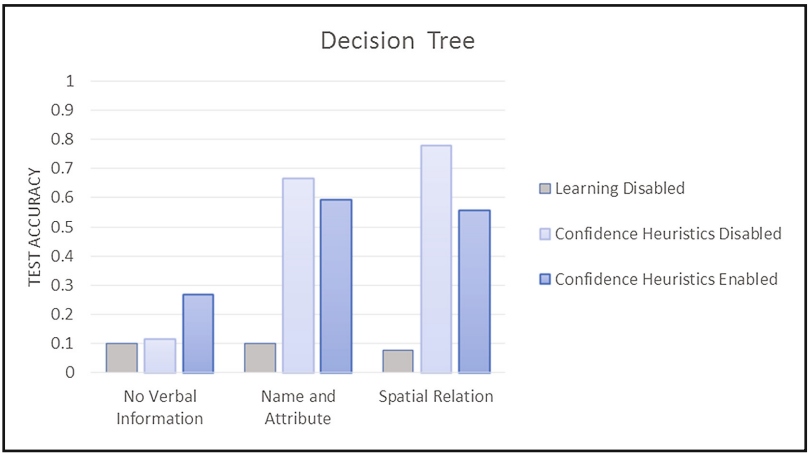

Fig. 5. Accuracy of the decision tree on the test dataset

complexity of the verbal information increased, the tendency to over-fit based on the confidence heuristics increased as well.

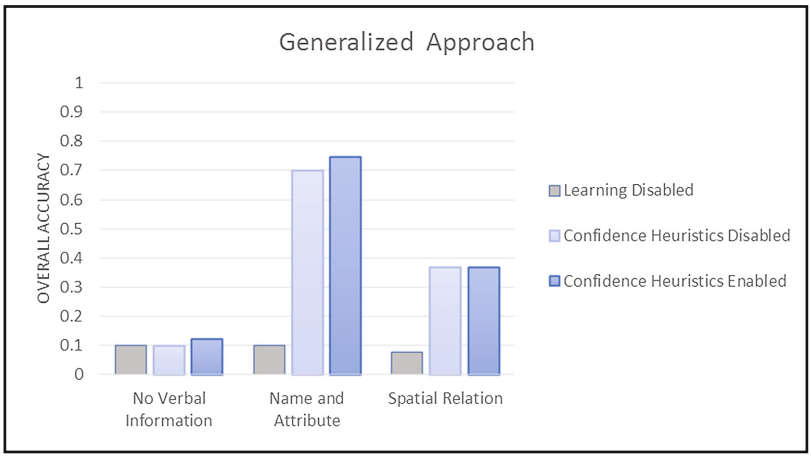

Fig. 6. Accuracy of the generalized approach on the entire dataset

The generalized approach was able to achieve accuracies of 10%, 70%, 37% for conditions (1), (2), and (3) when confidence heuristics were excluded. By including this metric, accuracies improved to 12%, 74% and 37% respectively. This model performed similarly well to the learning algorithms given simple verbal information but was unable to fully utilize more complex messages. Furthermore, the model was only able to take advantage of the heuristics to a small degree. This gives an insight to the difficulty of utilizing the abundance of information provided without the assistance of a learning algorithm.

5 Discussion

As seen by the results provided above, the SVM was able to achieve the highest overall accuracy. The DT, however, was unable to perform as well due to over-fitting. Additionally, the DT is limited to a max depth of 8 which forces the model to focus on key features and ignore some of the less significant ones. In comparison, the SVM can utilize as many features as desired so long as it benefits the classification process. Given this, the SVM is a suitable model for fusing multiple channels of information.

The generalized approach shows the challenge of extracting and making good use of the information available. However, the generalized approach achieved the highest accuracy when given a simple verbal command containing an object name and attribute. This result can be explained by the ease in which this information can be extracted, whereas spatial relations can take on many forms

and is harder to analyze. Failure to resolve part of the complex message can result in the information being diminished if not entirely lost.

By implementing a suitable learning algorithm we have shown that the combination of the hand pose and varying levels of verbal information can lead to improved reliability in the absence of other body pose information required to accurately resolve pointing gestures.

6 Future Work

Future improvements can be made by eliminating the need to preprocess the room. The AR markers used here simplify capturing object positions but could be replaced by segmenting objects in the Kinect depth stream in conjunction with the object localization provided by the SSD. Such a setup may require the use of multiple cameras in order to maximize the number of objects in view.

Additionally, the generalized approach could be improved for complex verbal messages by implementing a learning algorithm that transforms the verbal input into a set of observed features. If successful, the model could reach similar accuracies as the SVM while still being applicable to dynamic environments.

References

1. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., Berg, A.C.: SSD: single shot multibox detector. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 21–37. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_2
2. Abidi, S., Williams, M., Johnston, B.: Human pointing as a robot directive. In: 2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI), Tokyo, pp. 67–68 (2013)
3. Droeschel, D., Stckler, J., Behnke, S.: Learning to interpret pointing gestures with a time-of-flight camera. In: 2011 6th ACM/IEEE International Conference on Human-Robot Interaction (HRI), Lausanne, pp. 481–488 (2011)
4. Li, Z., Jarvis, R.: Visual interpretation of natural pointing gestures in 3D space for human-robot interaction. In: 2010 11th International Conference on Control Automation Robotics & Vision, Singapore, pp. 2513–2518 (2010)
5. Ueno, S., Naito, S., Chen, T.: An efficient method for human pointing estimation for robot interaction. In: 2014 IEEE International Conference on Image Processing (ICIP), Paris, pp. 1545–1549 (2014)
6. Shukla, D., Erkent, O., Piater, J.: Probabilistic detection of pointing directions for human-robot interaction. In: 2015 International Conference on Digital Image Computing: Techniques and Applications (DICTA), Adelaide, SA, pp. 1–8 (2015)
7. Pateraki, M., Baltzakis, H., Trahanias, P.: Visual estimation of pointed targets for robot guidance via fusion of face pose and hand orientation. In: 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops), Barcelona, pp. 1060–1067 (2011)
8. Stiefelhagen, R., et al.: Enabling multimodal humanrobot interaction for the Karlsruhe humanoid robot. IEEE Trans. Rob. **23**(5), 840–851 (2007)

9. Burger, B., Ferran, I., Lerasle, F., Infantes, G.: Two-handed gesture recognition and fusion with speech to command a robot. Auton. Robots **32**(2), 129–147 (2012)
10. Yamamoto, Y., Yoda, I., Sakaue, K.: Arm-pointing gesture interface using surrounded stereo cameras system. In: Proceedings of the 17th International Conference on Pattern Recognition, ICPR 2004, vol. 4, pp. 965–970 (2004)