

Contents

1 Learning Ownership

1

1 Learning Ownership

Ownagebot's comprehension and maintenance of ownership relations is facilitated by its object representation scheme. Each object is associated with a unique feature vector constructed upon Ownagebot's first visual scan of the object, and is updated in light of changes to the associated object upon subsequent scans. For the purposes of our experiments the feature vector included: the objects color, the distances between the object and the two avatars, the x , y , and z coordinates of the object relative to Ownagebot's base. Ownagebot also keeps track of its recent interaction history with objects, which includes: objects its attempted to pick up and, if the attempt was blocked, the label corresponding to the id of the avatar that applied the negative feedback, if any. As Ownagebot interacts with objects, older interactions are 'forgotten' in order to account for the possibility that older interaction memories no longer correspond to the more current state of ownership relations. Additionally, Ownagebot represents an object's ownership label as a list of probabilities summing to 1 that each of the encountered avatars (including Ownagebot itself) own the object.

Ownagebot actively learns ownership rules through its interaction with the environment. The basic learning can be broken down into three phases: *scanning*, *pickup*, and *feedback*. During the *scanning* phase, Ownagebot scans the environment, creating and updating a database of object feature vectors and avatar locations. It then enters the *pickup* phase, attempting to retrieve the object with the lowest probability of being owned by the other avatars, and bring it to the home area. During the pickup behavior, Ownagebot enters the *feedback* phase wherein it can receive negative feedback before completing the behavior. If negative feedback is administered, Ownagebot attempts to "offer" the object to each avatar, labeling the object in its interaction history with the id of the avatar that "accepted" the offer. Using the newly updated interaction history, Ownagebot updates the ownership probabilities of all encountered objects, and repeats this entire process until it has retrieved all allowable objects.

Initially, for each object, these probabilities are set to zero, save for Ownagebot's ownership probability, to facilitate exploratory interactions with objects in the environment. These probabilities are updated as a function

of how similar the object is to the objects in its interaction history. Here the similarity between an object o and a previously manipulated object c is calculated using the real-valued radial basis function

$$\phi(o, c) = e^{-\frac{1}{2}(\frac{r}{\sigma})^2}$$

Where σ is a constant denoting the variance of the Gaussian distributions and r is the weighted euclidean distance between feature vectors v_o and v_c corresponding to objects o and c respectively

$$r = ||\alpha^T(v_o - v_c)||^2$$

where α is a vector weighing each dimension of the feature vector. Given that r a function of euclidean dist

Objects that are similar to an object in the interaction history assume a higher probability of sharing its ownership label. As objects are added to the interaction history and older objects are removed, these probabilities are recalculated to keep in step with the dynamic nature of ownership relations. More specifically, the probability that an avatar A_j owns an object o_i given a set of previous interactions H at time $k + 1$ is calculated according to the following equation

$$p_i(A_j)^{k+1} = (1 - \lambda)p_i(A_j)^k + \lambda \frac{\sum_{c_j \in H} \phi(o_i, c_j)}{\sum_{\forall c \in H} \phi(o_i, c)}$$

Here, $\lambda \in [0, 1]$ weighs how much the previously determined probability affects the current calculation.