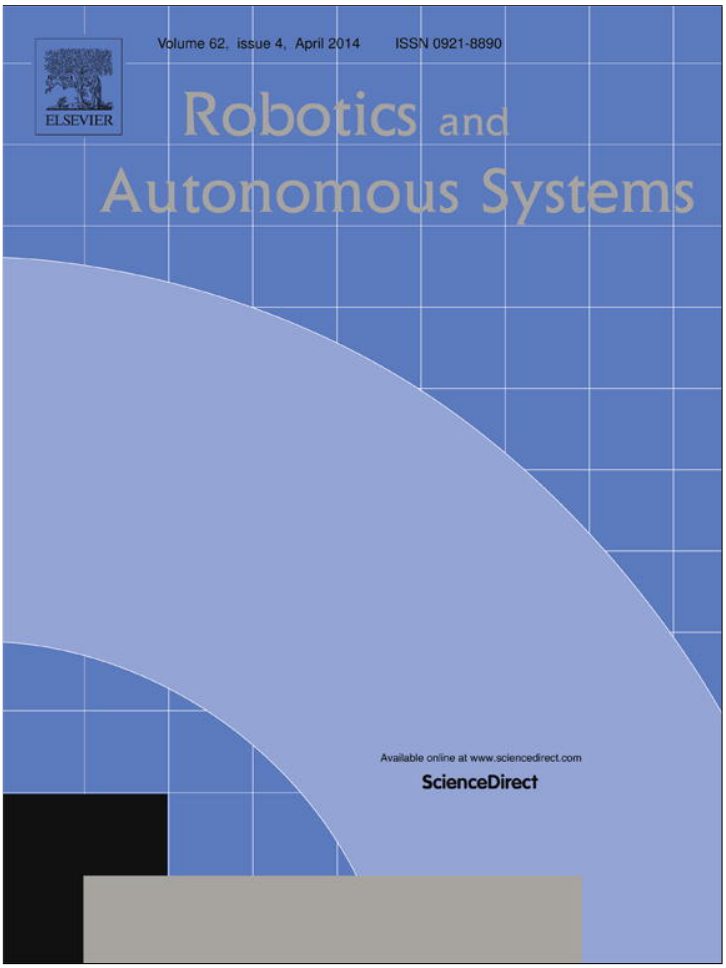

Robotics and Autonomous Systems 62 (2014) 487–496

Contents lists available at ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot

Object–object interaction affordance learning

Yu Sun *, Shaogang Ren, Yun Lin

Department of Computer Sci & Eng, University of South Florida, 4202 E. Fowler Ave, Tampa, FL 33613, United States

ARTICLE INFO

Article history:
Received 7 April 2012
Received in revised form
10 October 2013
Accepted 9 December 2013
Available online 14 December 2013

Keywords:
Action recognition
Robot learning
Learn from demonstration
Object classification
Graphical model

ABSTRACT

This paper presents a novel object–object affordance learning approach that enables intelligent robots to learn the interactive functionalities of objects from human demonstrations in everyday environments. Instead of considering a single object, we model the interactive motions between paired objects in a human–object–object way. The innate interaction-affordance knowledge of the paired objects are learned from a labeled training dataset that contains a set of relative motions of the paired objects, human actions, and object labels. The learned knowledge is represented with a Bayesian Network, and the network can be used to improve the recognition reliability of both objects and human actions and to generate proper manipulation motion for a robot if a pair of objects is recognized. This paper also presents an image-based visual servoing approach that uses the learned motion features of the affordance in interaction as the control goals to control a robot to perform manipulation tasks.

1. Introduction

Object categorization and human action recognition are important capabilities for an intelligent robot. Traditionally, these two problems are treated separately. However, manipulation skills and object affordance are highly related for humans. Therefore, seeking an approach that can connect and model the motion and features of an object in the same frame is considered a new frontier in robotics. With the boom in learning from demonstration techniques in robotics [1–3], more and more researchers are trying to model object features, object affordance, and human action at the same time. Most of the work builds the relationship between single object features and human action or object affordance [4–6].

In daily life, when we are performing tasks, we pay most of our attention to object states or object interactions. For example, when we are writing on paper with a pen, we focus our attention on the pen point, which is the interaction part between the pen and the paper. Moreover, object interaction can directly reveal an object's functions. For instance, when we put a book into a schoolbag, the putting motion tells us that the schoolbag is a container for books. There are endless interactive examples with paired objects in our daily lives. Fig. 1 shows several objects on a table that have an inter-object relationship: a CD and a CD case, a pen and a piece of paper, a spoon and a cup, and a cup and a teapot. In this paper, we attempt to capitalize on the strong relationship between paired objects and interactive motion by building an object relation model and associating it with a human action model in the human–object–object way to characterize inter-object affordance.

The interactive motions of these objects are better defined if we know the interactive pairs. For example, in daily life, we move a teapot in many different ways, such as putting it on a table, storing it on a shelf, and washing it. However, if we have a teapot and a teacup in a scene, water-pouring motion is more likely to occur. Likewise, if we recognize a pouring motion and a teacup, it is very likely that the object associated with the pouring motion is the teapot. We define the interactive motion between paired objects as the object–object-interaction affordance that is connected to both objects. Object–object-interaction affordance is not only useful for object and motion recognition, but also important for robotic learning, as robots can learn object–object-interaction affordance as a manipulation skill that is intrinsic to the paired objects.

Object affordance cognition is one of many core capabilities that a robot needs to gain before it can intelligently perform tasks in the real world. However, this challenging problem has been explored only recently in limited works. Many of the current works model object affordance with interaction between a single object and human action and then use the mutual relationship to improve the recognition of each other. Gupta and Davis [4] recently achieved inspiring success in using single object–action to improve the recognition rate of both the object and human motion. Jiang et al. [7] encoded human preferences about object placements along with the geometric relationship between objects and their placing environments. Kjellstrom et al. [5] used conditional random field (CRF) and factorial conditional random field (FCRF) to model the object type and human action relationship and estimated the 3D hand pose to represent human action, which includes open, hammer, and pour actions. Yao and Li [8] modeled the mutual

* Corresponding author. Tel.: +1 8139747508.
E-mail address: yusun@cse.usf.edu (Y. Sun).

0921-8890/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.robot.2013.12.005

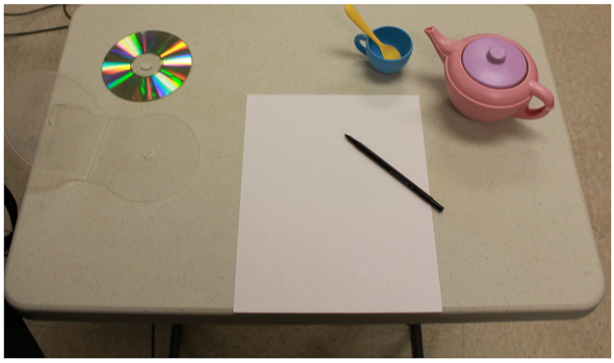

Fig. 1. Several objects on a table have inter-object relationships.

context information between human poses and objects in still images using a structure-learning method to model the human and object interaction and achieved a state-of-the-art result in object and human pose detection in static images. Most recently, Gall et al. [6] recovered human action from depth images and used it to represent object function and affordance. In their work, objects were classified according to the involved human action in an unsupervised way based on high-level features.

Some recent works have tried to infer object affordance from object low-level features or 3D shapes. Stark et al. [9] obtained object affordance cues from human hand and object interaction in training images and then detected an object and determined its functions according to its affordance cue features. Grabner et al. [10] proposed a novel way to determine object affordance using computer graphical simulation. With 3D object shapes, their system "imagines" an actor performing actions on objects in a scene to determine the objects' affordances. In [10], first the 3D geometry of a single indoor image was recovered, and then the affordances of the objects were inferred from the joint space of human poses and scene geometry by modeling the physical interaction.

In the robotics community, several works obtained and used object–action relation without considering many low-level object features. In [11], concrete object recognition was not considered, and objects were categorized solely according to object interaction sequences. Objects were segmented out from a number of video sequences, and an undirected semantic graph was used to represent the space interaction relationship between objects. With a sequence of graphs, their work was able to represent object temporal and spatial interactions in an event. With the semantic graphs, they constructed an event table and a similarity matrix. The similarity between two sequences of object interaction events could be obtained according to the similarity matrix. The objects could further be categorized according to their roles in the interactions, and the obtained semantic graphs might be used to represent robotic tasks.

In summary, most current works focus on object–action interaction or low-level object affordance features. Few investigate the affordance relationship between objects. This paper presents a way to model inter-object affordance and then use the inter-object affordance relationship to improve object and action recognition.

Studies in neuroscience and cognitive science on objects' affordance [12] indicate that the mirror neurons in human brains congregate visual and motor responses [13–15]. Mirror neurons in the F5 sector of the macaque ventral premotor cortex fire both during observation of interacting with an object and during action execution, but do not discharge in response to simply observing an object [16,17]. Recently, Yoon et al. [18] studied the affordances associated to pairs of objects positioned for action and found an interesting so-called "paired object affordance effect". The effect was that the response time by right-handed participants is faster if the two objects were used together when the active object (supposed to be manipulated) was to the right of the other object. Borghi et al. [19] further studied the functional relationship between paired objects and compared it with the spatial relationship and found that both the position and functional context are important and related to the motion; however, the motor action response is faster and more accurate with the functional context than the spatial context. The study results in neuroscience and cognitive science indicate that there are strong connections between the observation and the motion, and functional relationships between objects are directly associated with the motor actions. A comprehensive review of models of affordances and the canonical mirror neuron system can be found in [20].

Inspired by the studies above, we propose to capitalize on the connection between the observation of functional-related objects and active functional motion actions to address the skill-learning problem in robotics. In this paper, we simplify the functional-related objects with piece-wise functional-related paired objects and model the inter-object manipulation motions as the inter-object affordance and associate it with paired-object recognition. The goal is to allow robots to learn inter-object affordance motions from humans and then trigger the robot to generate the correct manipulation motion when observing the paired objects.

To model the functional relationship of the paired objects and their relationship with the manipulation motion, this paper presents a graphical model that connects the paired objects and the manipulation motions. The graphical model intuitively represents the functional connectivity of the objects, such as a teapot and a cup or a book and a schoolbag, and extends that connectivity to manipulation motions. A Bayesian Network is employed to model these relationships, in which the paired objects, the interact action, and the consequence of the object interaction are included as a node in the graphical model.

In addition, we developed a method to recognize the paired objects and human motion by analyzing the interactive motion and the statistical knowledge learned from training data. We also constructed a method to leverage object recognition accuracy from videos with the recognition of human interactions, and vice versa. With hand motion trajectory and statistical knowledge learned from training data, the detection accuracy of the interactive objects is significantly improved. With the recognition of the objects, the interactive motions carried out by humans are recognized with much higher accuracy as well.

The interactive motions associated with the paired objects can be learned as the affordance in interaction with statistical models such as Gaussian mixture models. The learned motion can then be directly used to control a robot to perform the proper manipulation motion when the robot sees the paired objects. This paper presents an image-based visual servoing approach that uses the learned motion features in the interaction affordance as the control goals to control the robot to perform the manipulation task instead of manually programming the motion.

We recruited 6 subjects, evaluated our approach with 5 pairs of objects in experiments, and recorded the interactive motion in 50 video sequences.

2. Model human–object–object-interaction affordance

Fig. 2 illustrates the workflow of our framework. We first obtained the initial likelihood of the objects' manipulation and reaction. The object initial likelihoods were estimated with a sliding window object detector, which is based on the Histogram of Oriented Gradients (HoG). We estimated the initial likelihood of human action based on the feature of human hand motion trajectory. The human hand was tracked in the whole process, and the hand motion was segmented according to the velocity changing. With motion segmentation and possible object locations, the interactive

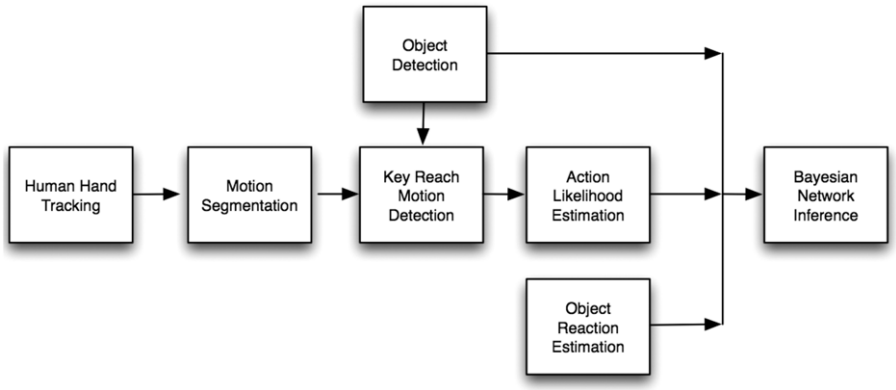

Fig. 2. Framework workflow.

object pairs were detected in the step of key reach motion detection. The start time of the manipulation was estimated based on the object pair locations and hand motion trajectory. Then, the initial belief of the manipulation was computed.

Object interaction usually leads to a state change of the associated objects. For example, if a CD is put into a CD case, the color of the CD case probably will change. The likelihood of object reaction was estimated by comparing with the training datasets. Finally, the belief in each node was updated with the inference algorithm for Bayesian Networks.

2.1. Bayesian network for human–object–object affordance

A Bayesian Network is a powerful inference tool for decision making in the observation of several or many interrelated factors. The belief for each node can be updated with messages from other evidence nodes. In our Bayesian Network (as illustrated in Fig. 3), the two interactive objects are represented as O_1 and O_2. M denotes hand manipulation motion, and O_R is the object reaction, which denotes the object state change after the interaction. The inter-object affordance, which is also the human action or manipulation M, is determined by the two interacting objects (O_1 and O_2), and they are the parents of node M. Similarly, the object reaction is the consequence of the two objects and the manipulation, so it becomes the child of the three nodes in the graph. The remaining nodes are evidence, $e = \{e_{O_1}, e_{O_2}, e_M, e_{O_R}\}$, and they represent the evidence for O_1, O_2, M, and O_R, respectively. Using the Bayesian rule and conditional independence relations, the joint probability distribution can be represented with Eq. (1). After we obtain the evidence, we can estimate the belief of each node with loopy believe propagation algorithms. Each item in the right side of Eq. (1) is discussed in the following sections.

$$\begin{aligned} P(O_1, O_2, M, O_R|e) \propto\ & P(O_1|e_{O_1})P(O_2|e_{O_2}) \\ & P(M|O_1, O_2)P(M|e_M) \\ & P(O_R|O_1, O_2, M)P(O_R|e_{O_R}) \end{aligned} \quad (1)$$

Our Bayesian model can be scaled up by increasing the number of variables for object and action in each node without changing the graphical model structure. Alternatively, we can combine multiple Bayesian networks to form a large-scale graphical model if there are inter-connections between different pairs of objects.

2.2. Object detection

To estimate the initial likelihood of the objects, we used an approach similar to [21]. The detector works in the sliding window manner, and we used a variant of the HoG feature from [22] to represent the object local features. At each pixel, the color channel

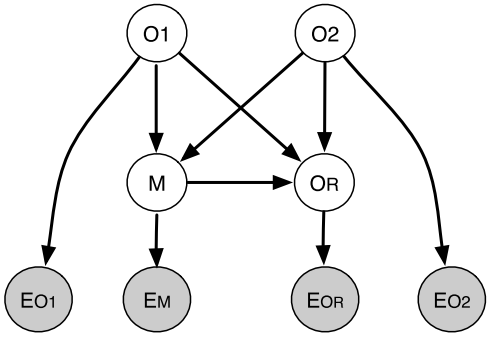

Fig. 3. Graphical model for motion and object interaction. O_1 and O_2 represent the two interacting objects, M denotes hand manipulation motion, and O_R is the object reaction.

with the largest gradient magnitude was used to represent the gradient orientation and magnitude. In each detecting window, the image was divided into 8×8 pixel cells and, for each cell, the pixel level feature was aggregated to a feature map.

We collected our training images from the Image-Net [23] and Google Image Search. All of the training images were labeled. For each object, we used around 50 positive and 70 negative examples to train an SVM (Support Vector Machine) classifier. The window size and aspect ratio were learned from the training dataset. For each object class, we trained a bi-class classifier. The LibSVM library [24] was used to obtain the probability of the classification for each window.

Objects were modeled as object type and object location. We computed the object likelihoods:

$P(O_1 = \{obj_1, l^{O_1}\}|e_{O_1})$ and

$P(O_2 = \{obj_2, l^{O_2}\}|e_{O_2})$ for each sliding window with the SVM estimation. Fig. 4 shows a sample of the detection results.

2.3. Motion analysis

The object detector gave us only the possible object locations. To detect two objects locations involved in an action and estimate the initial likelihood of the manipulation, we analyzed the hand motion. After the hand motion tracking, the motion trajectory was segmented into several pieces. The likelihood of the motion type was estimated based on the motion segments. Generally, there are two kinds of object interactive motion – putting an object into a container and manipulating one of the objects relative to the other [25,26]. In this work, we did not discriminate these two kinds of motion, although they are considered different in cognition science.

Fig. 4. Example result of object detection with SVM classifier using HoG features. Dots indicate detected object centers.

2.3.1. Human hand tracking

Since the inter-object affordance is represented by the object motion interaction that is controlled by hand motion, the hand motion needs to be tracked to model the interaction. It is difficult to track a hand based on shape information because of the high motion speed and the variability of the hand gesture. However, human skin color is a very stable feature that can be used to track human activities [27]. We tracked the hands by combining the skin color model [28] and the TLD object tracker [29]. The hand was initially located using optical flow and skin color in the initial several frames. Then, in each frame, the hand location was refined according to the color information around the previous hand location and the shape features from the TLD tracker. Only the right-hand motion was tracked in our experiments. There are other 2D hand-tracking methods, such as the one in [30], which uses the skeleton of the human upper body. Fig. 5(b) shows the tracked trajectory.

2.3.2. Motion segmentation

After getting the hand motion trajectory, the trajectory was segmented into several pieces according to the magnitude of the motion velocity. There are two modes for human limb motion: ballistic and mass spring motion [31]. The ballistic motion represents the motion that starts with long acceleration and ends with long deceleration, such as reaching an object. The mass spring motion is the motion with several accelerations and decelerations, or the motion that has very low velocity. First, the trajectory was segmented into small pieces by local minimal points, and then these pieces were merged or segmented into possible ballistic and mass spring segments. Similar to the method in [31], the segments were classified into ballistic and mass sprint types according to the speed feature. The features used include maximum velocity, average velocity, number of local minimum points, standard deviation, motion distance, etc. Fig. 5 (c) gives the segmented velocity for one motion of putting a pencil into a pencil case. Similar motion analysis approaches exist in neuroscience and cognitive science to classify and represent motion segments with action chains [32,33].

2.3.3. Key reach motion detection

In each object interaction process, a human hand carries one object to the location of another object. For example, in a stirring water example, the spoon needs to be moved to the cup. We represent this reach motion as the key reach motion. There could be several reach motions in one action. For example, if we want to put a book into a schoolbag, we need to first open the schoolbag, which is the first reach motion; reach to the book, which is the second reach motion; and take the book to the schoolbag and put into it, which is the third reach motion. The key reach motion here is to take the book to the schoolbag. Therefore, we named the book as the start object and the schoolbag as the end object. In our graphical model, book was represented as object1 and schoolbag as object2. We wanted to detect the key reach motion and the interacting object pair at the same time once we obtained the detected objects and hand motion trajectory.

The ballistic segments were classified into reach motion and non-reach motion according to acceleration and deceleration velocity and time duration, average velocity and standard deviation of speed, etc. It was difficult to determine the key reach motion based only on hand motion information, and it was also difficult to detect whether the hand was carrying an object or not if the object was small. But it was easy to detect the object state around the start and end location of the reach motion. The key reach motion started from one location (l^M_{r1}) and ended at another location (l^M_{r2}). The distance between the location of the start object (l^{O_1}) and l^M_{r1} was modeled as a normal distribution, which was $N(|l^M_{r1}\, l^{O_1}|; \mu_r^{O_1}, \sigma_r^{O_1})$. The distance between the location of the end object (l^{O_2}) and l^M_{r2} was also modeled as a normal distribution, which was $N(|l^M_{r2}\, l^{O_2}|; \mu_M^{O_2}, \sigma_M^{O_2})$. The start and end locations for each reach motion were known. The start object, end object, and key reach motion were detected at the same time, according to the two distribution values. Here, $\mu_r^{O_1}$, $\sigma_r^{O_1}$, $\mu_M^{O_2}$, and $\sigma_M^{O_2}$ were learned from the training data set. In the key reach motion, the human hand needs to carry object1 from location l^{O_1} to l^{O_2}, so the key reach motion can be further ensured by checking whether the detected start object is removed or not. This can be done by comparing the likelihood value of object1 at location l^{O_1} before and after the key reach motion. Fig. 6 shows the key reach motion segment detected (marked as red) from the entire motion that put a pencil into a pencil case.

2.3.4. Manipulation motion estimation

Human hand trajectory was used to estimate the likelihood of manipulation motion. The manipulation motion was modeled with five parameters: start time (t^M_s), end time (t^M_e), two reach locations (l^M_{r1}, l^M_{r2}), and manipulation type (T^M). According to Eq. (1), we needed to model the conditional probability $P(M|O_1O_2)$ and the initial likelihood value for M, $P(M|e_M)$.

The term $P(M|O_1O_2)$ was modeled in Eq. (2). Let l^M_s be the hand location for the start time t^M_e, we can model $P(t^M_s, t^M_e|O_1O_2)$ as $N(|l^M_s l^O|, \mu_r^O, \sigma_r^O)$, and O can be O_1 or O_2. μ_r^O is the mean grasping distance for object O, and σ_r^O is the variance. Both of them can be learned from the training data. $P(l^M_{r1}|O_1)$ and $P(l^M_{r2}|O_2)$ are modeled as normal distributions $N(|l^M_{r1}l^{O_1}|, \mu_r^{O_1}, \sigma_r^{O_1})$ and $N(|l^M_{r2}l^{O_2}|, \mu_M^{O_2}, \sigma_M^{O_2})$, which were discussed in Section 2.3.3. $P(T^M|obj_1, obj_2)$ was computed according to the occurrence of manipulation type and object type in the training data.

$$P(M|O_1O_2) = P(t^M_s, t^M_e|O_1O_2)P(l^M_{r1}|O_1)P(l^M_{r2}|O_2) \times P(T^M|obj_1, obj_2). \quad (2)$$

We estimated the likelihood $P(M|e_M)$ with the features from the hand motion trajectory. Based on the segmentation in Section 2.3.2, theballistic and mass spring segments were replaced with labels. The manipulation motions were classified according to the number of ballistic and mass spring segments, translation rate of the two segments, time duration, etc. Linear SVM was trained as the classifier and gave the likelihood of the manipulation.

2.4. Object reaction

Object reaction was modeled with two parameters, reaction type (T^R) and reaction location (l^R). It was difficult to model the

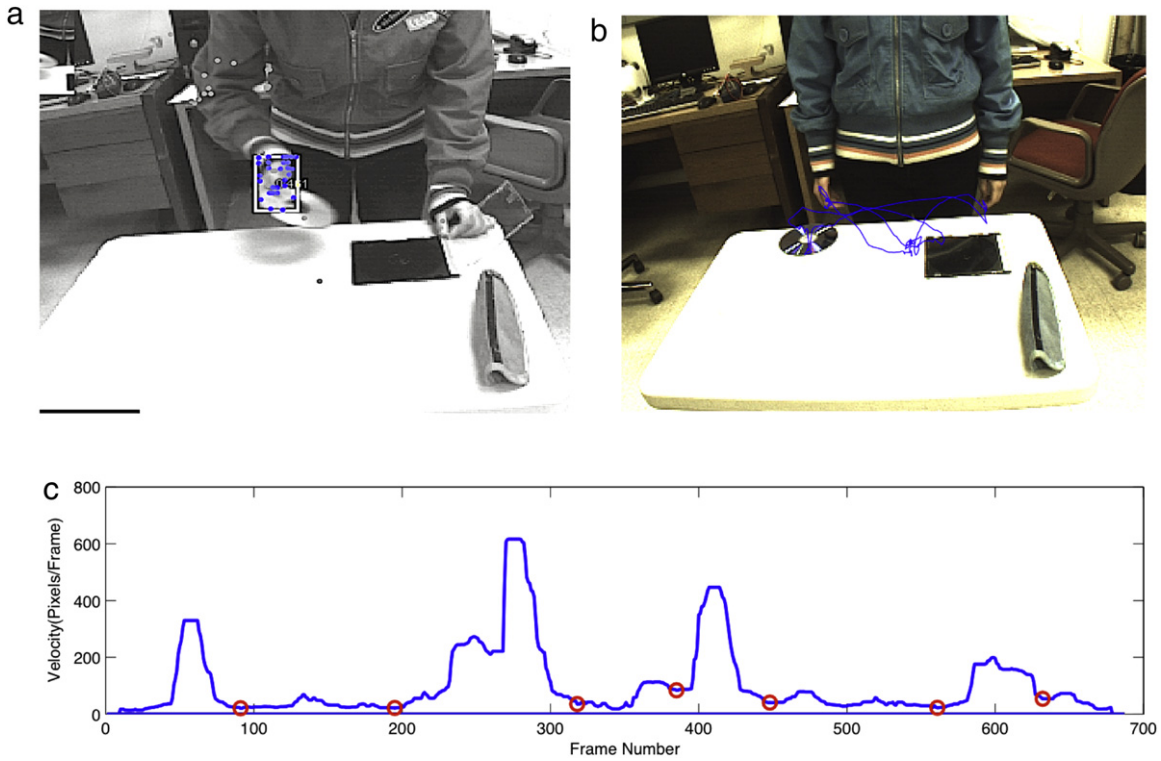

Fig. 5. Hand tracking and motion segmentation: (a) right-hand motion tracking; (b) right-hand motion trajectory; (c) motion segmentation with velocity—horizontal axis is time (frame number), and vertical axis represents velocity (pixels per frame). Red circles are detected motion segment boundaries.

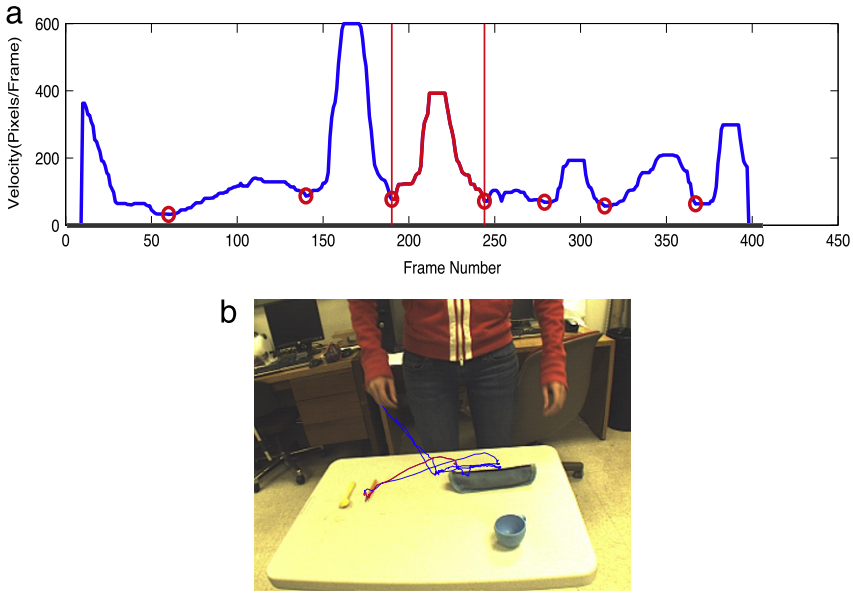

Fig. 6. Key reach motion detection: (a) red velocity segment represents key reach motion in velocity graph. The red circles are detected motion segment boundaries; (b) The red curve shows key reach motion in image.

object reaction since the object states can change in many possible ways, such as in color and shape. Here, we considered only the state change of object2 after the interaction. Similar to [4], we used the color histogram change around object2 to represent the object reaction. We estimated $P(O_R|e_{O_R})$ by comparing the histogram change with the training instances from the training dataset. We modeled the prior $P(O_R|O_1, O_2, M)$ according to Eq. (3). The item $P(l^R|O_2)$ is model as $N(|l^R l^{O_2}|; \mu^R, \sigma^R)$, and parameters μ^R and σ^R were learned from the training data. The $P(T^R|O_1, O_2, M)$ was learned from the training dataset by counting the occurrence of T^R, O_1, O_2 and M as shown in the following:

$$P(O_R|O_1, O_2, M) = P(l^R|O_2)P(T^R|O_1, O_2, M) \quad (3)$$

2.5. Bayesian network inference

After getting the key reach motion and the interactive object pair locations, we estimated the parameters for M and O_R according to Sections 2.3.3 and 2.3.4. We did the inference with a loopy

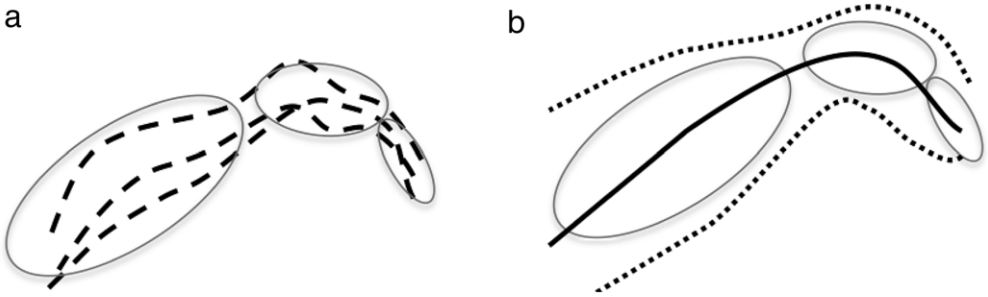

Fig. 7. (a) A set of trajectories modeled with three Gaussian distributions; (b) The desired trajectory is generated from the Gaussian distributions with Gaussian Mixture Regression (GMR).

believe propagation algorithm [34] once all of the initial likelihoods for O_1, O_2, M, and O_R were estimated. The Bayesian Network, the object classifier, and the manipulation classifier were trained with fully-labeled data.

3. Interaction affordance based visual servoing

Visual servo control provides an elegant approach to control a robot directly with 2D or 3D visual feedback due to its simplicity and robustness. Several standard approaches have been described in great detail in tutorials such as [35,36], in which two main forms of visual servoing exist. Here, we mainly focused on image-based visual servoing (IBVS) where velocity control signal is computed based on the 2D errors in features in the image. Position-based visual servoing (PBVS) is another form that relies on 3D positions of feature points in the image which can be applied with 3D sensors such as a PrimeSense sensor. The IBVS essentially uses the error between matched features in the pre-defined goal image and the current image to compute a velocity control signal and then uses the signal to control the robot towards the desired pose. The visual servo technique has been successfully implemented in many applications. Our previous work [37] designed an IBVS approach to use an eye-in-hand monocular camera for combined control of mobility and manipulation for the 9-DoF WMRA system (7-DoF robotic arm and a 2-DoF power wheelchair platform) to execute activities of daily living (ADL) autonomously.

Similarly, the learned affordance in interaction, which is associated to the paired objects, can be directly used to control a robot to perform the proper manipulation motion. Instead of programming the robot to perform the manipulating task manually, we designed an image-based visual servoing approach that uses the motion features in the affordance as the control goals. To simplify the problem, we considered only one-hand manipulation, and one of the paired objects was stationary. We further assumed that the relative position between the camera and the stationary object was fixed during the learning and the robot manipulation. The assumption is usually satisfiable in a regular learning from demonstration framework by using the same fixed camera in learning and robot manipulation.

From multiple observations of the same task, we could further model the key reach motion (Section 2.3.3) with several motion states with Gaussian mixture model (GMM). As illustrated in Fig. 7, a set of training key reach motion trajectories can be modeled with a number of Gaussian distributions by the set of parameters $\{\pi_k, \mu_k, \Sigma_k\}$, where k is the index of the distributions, π_k is the prior probability, μ_k and Σ_k are the center and covariance matrices of the kth distribution respectively.

To apply visual servoing, we used the motion of the image features to represent the relative object–object motions. Each of the features was tracked and modeled with GMM. After the GMMs are trained with multiple demonstrations, a robot can generate a series of target image features (feature trajectories) from the learned GMMs with Gaussian Mixture Regression (GMR) [38] and use desired image feature trajectories as the control signal to guide the manipulator to perform the learned interactive motion.

4. Experiments and results

We evaluated our framework with a dataset collected from 6 subjects who performed 5 interactions. The data of 4 subjects was used for training, and the data of the other 2 subjects was used for testing. Each subject performed each action for 2 or 3 trials. The interaction object pairs included teapot–cup, pencil–pencil case, bottle cap–bottle, CD–CD case, and spoon–cup. The actions for these object pairs were pouring water from a teapot to a cup, putting a pencil into a pencil case, screwing on a bottle cap, putting a CD into a CD case, and stirring a spoon in a cup. All of these objects and actions were chosen because they are very common in everyday life and are representative of different inter-object affordance relationships. In addition to evaluating the performances of object and action recognition with inter-object affordance, we also demonstrated the proposed affordance-based visual servoing approach with our FANUC robotic arm and Barrett hand.

4.1. Training data

We trained the object classifier, the action classifier, and the Bayesian Network in a supervised manner. For the object classifier, the training images were collected from the ImageNet [23] and Google Image Search. For the action classifier and the Bayesian Network, the training data were collected from manually-labeled video sequences. About 50 video sequences that were performed by 4 subjects were used for training. For each training video sequence, object locations, reach locations, and action type and the start frame of the manipulation were labeled. Fig. 8 gives the velocity-changing example for each action. Fig. 8(a–e) shows the recorded motions of putting a CD into a CD case, putting a pencil into a pencil case, pouring water from a teapot to a cup, screwing on a bottle cap, and stirring water in a cup, respectively.

4.2. Object classification

The test dataset contains the action sequences performed by two subjects. Fig. 9 shows the object classification confusion matrices for object1, which was located at the beginning of the key reach motion. Fig. 10 gives the likelihood confusion matrices for object2 located at the end of the key reach motion. For each confusion matrix, the ith row represents the likelihood value when the ith type of object is present. For object1, it was difficult to distinguish pencil and spoon based on their appearances because they have a similar shape and both are small. Within the context of human–object–object interaction, we can see that the spoon and pencil can be distinguished accurately. The recognition rate for object1 improved from 72.6% to 86%, and the recognition rate for object2 improved from 75.3% to 82.8%.

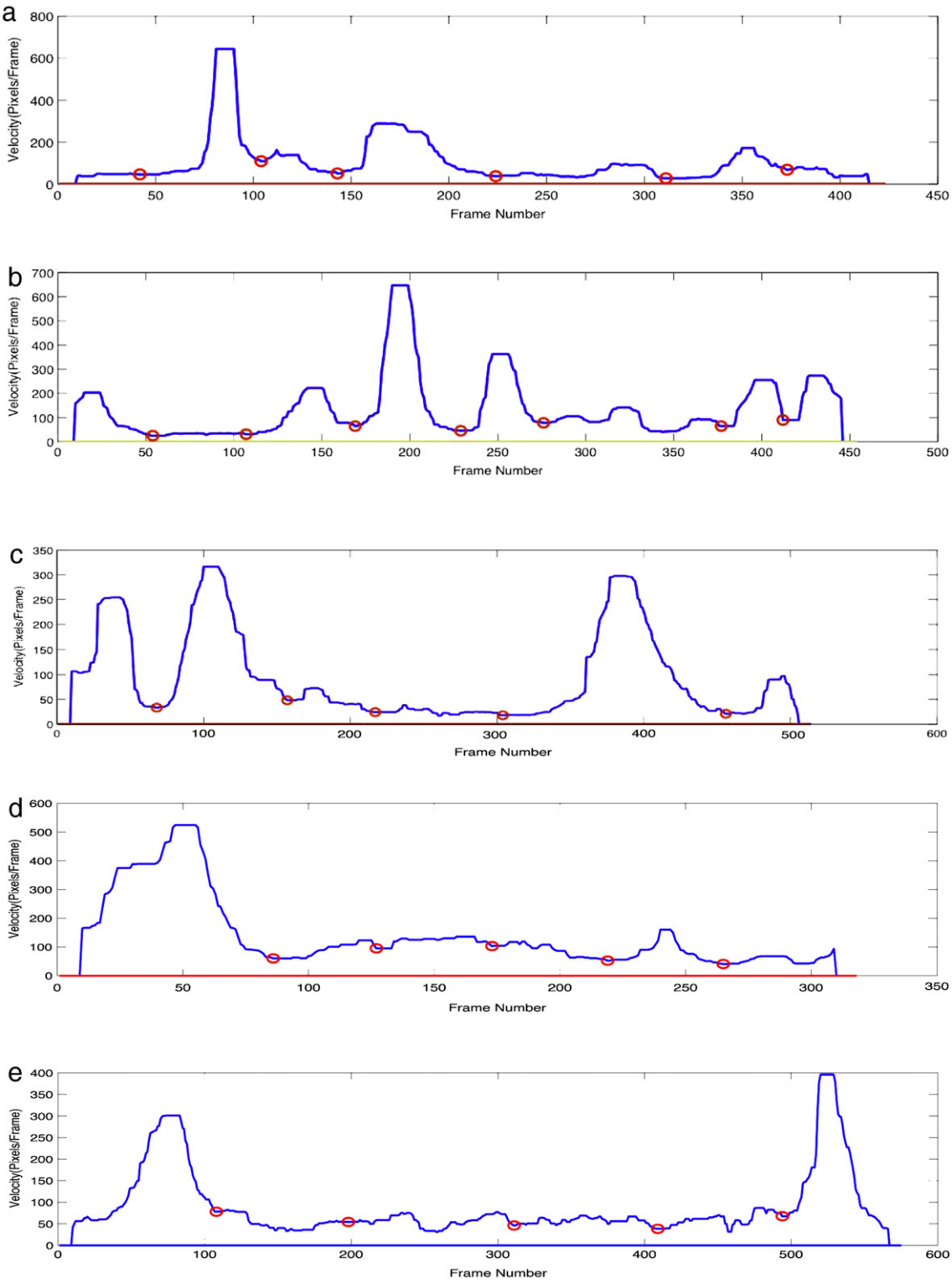

Fig. 8. Motion velocity diagram for five actions. Horizontal axis is time (frame number), and the vertical axis represents velocity (pixels per frame). (a–e) show the recorded motions of putting a CD into a CD case, putting a pencil into a pencil case, pouring water from a teapot to a cup, screwing on a bottle cap, and stirring water in a cup, respectively.

4.3. Action recognition

Among the five actions studied, if based only on motion features, it was difficult to distinguish putting a CD into a CD case, putting a pencil into a pencil case, pouring water into a cup, and stirring water in a cup because they have the similar motion patterns. With the human–object–object interaction framework, they can be distinguished. Fig. 11(a) shows the likelihood confusion matrix that is estimated with only hand motion features. Fig. 11(b) shows the action confusion matrix using human–object–object interaction framework. We can see that the overall average recognition rate across all objects improved from 42.6% to 83.0%.

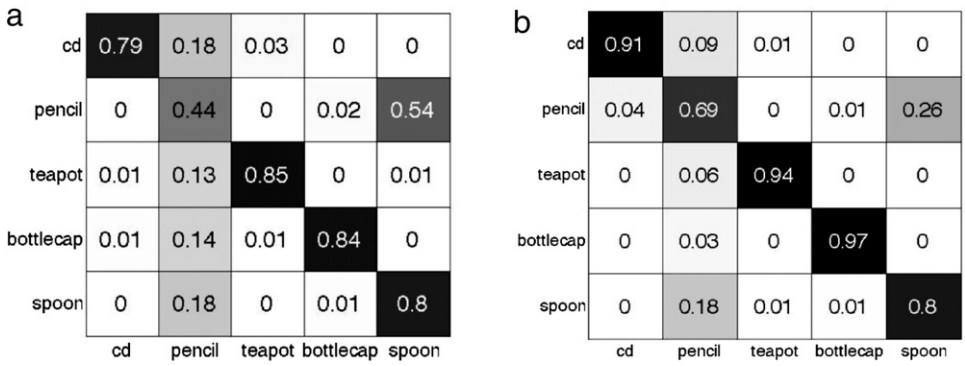

Fig. 9. Object1 likelihood confusion matrix. (a) shows the result using the HoG detector, (b) shows the result using framework.

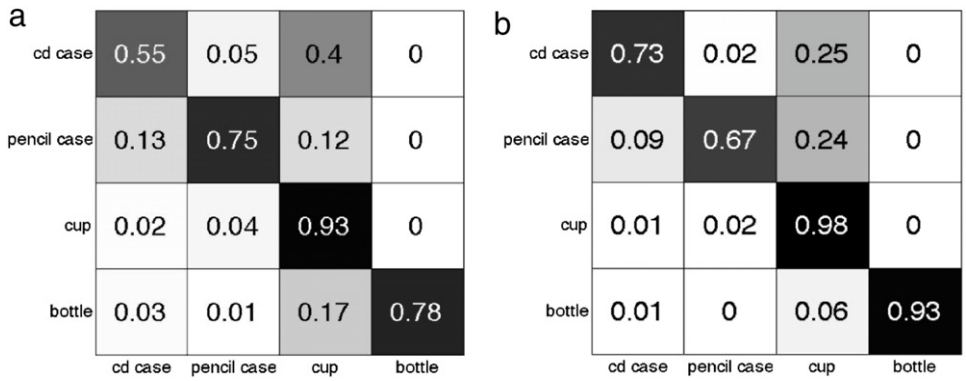

Fig. 10. Object2 likelihood confusion matrix. (a) shows the result using the HoG detector, (b) shows the result using framework.

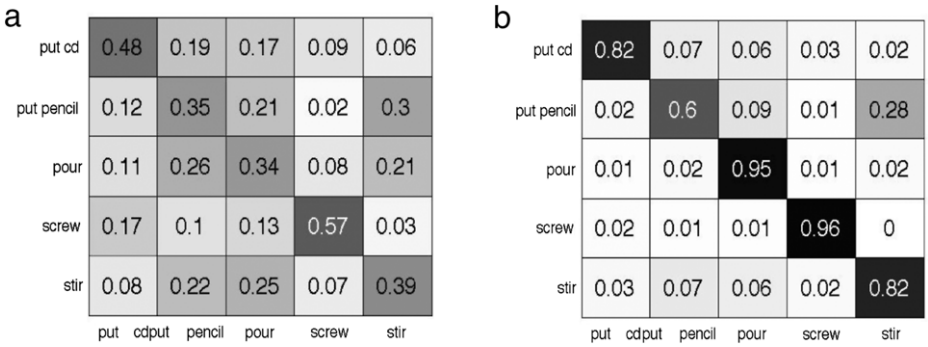

Fig. 11. Action likelihood confusion matrix: (a) result using only motion features; (b) result using framework. The *i*th row shows likelihood value when *i*th action is categorized.

4.4. Interaction affordance visual servoing

In addition to leveraging the motion and object recognition accuracy, the learned affordance in interaction, which is associated to the paired objects, can be directly used to control a robot to perform the proper manipulation motion. Instead of programming the robot to perform the manipulating task manually, we designed a visual servoing approach that used the motion features in the affordance as the control goals. The experiment setup is shown in Fig. 12. A Point Grey Flea video camera was used as the visual feedback, and our visual servo controller controlled a 6-DOF Fanuc L200IC robotic arm and a Barrett robotic hand to perform the learned manipulation motions.

In our experiment, as discussed in Section 3, due to the resolution limitation of our camera, it was very difficult to obtain enough robust visual features on all the interactive objects. However, various structural features, such as contour and edges, could be used, which is beyond the topic of the paper. To demonstrate the approach without loss of generality or diverting from the main idea of the paper, we used the color information of both objects to segment them from the background and used their centers to perform a simplified position-based visual servoing.

Fig. 13(a) shows the training 2D trajectories of the teapot in its key reach motion for its interaction with a cup (relative motion to the center of the cup). From the five trials observed, the key reach motion could be modeled with three Gaussian distributions, as shown in Fig. 13(b). The models of the Gaussian distributions were trained to represent the interactive manipulation motion of the teapot when associated with the teapot–cup pair.

After the learning phase, when our robot observed a teapot and a cup, it looked up the stored manipulation models associated with the pair of objects and retrieved the affordance in interaction modeled with Gaussian distributions and then used the models to generate the desired manipulation it should perform with GMR. Fig. 13(c) shows the generated 2D trajectory that was used for visual servoing input to control the robot to perform the desired manipulation.

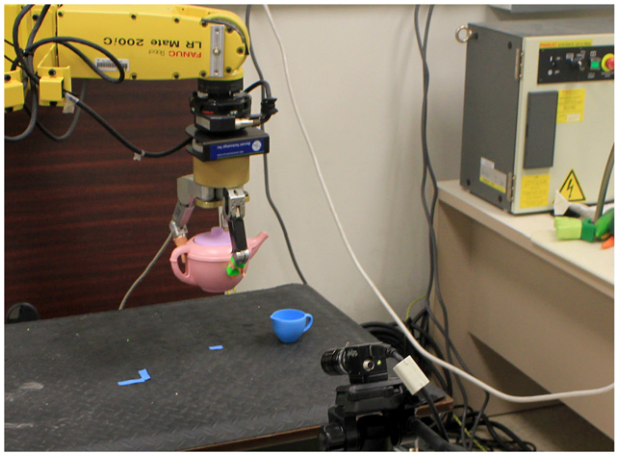

Fig. 12. A visual servoing setup with a 6-DOF Fanuc L200IC robotic arm, a Barrett robotic hand, and a fixed Flea camera.

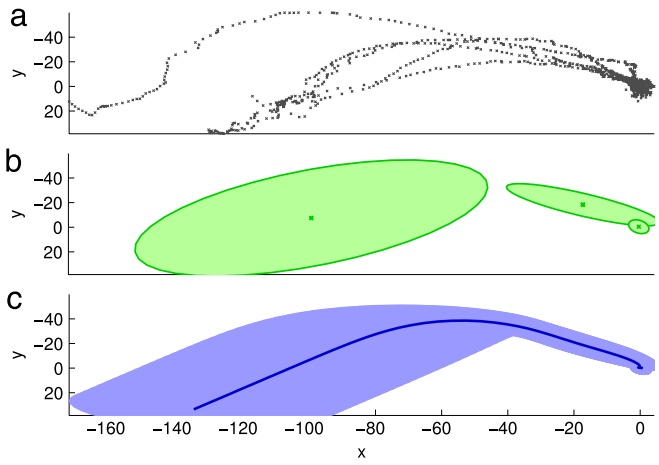

Fig. 13. (a) The training 2D trajectories of the pouring motion associated to a teapot and a cup; (b) The Gaussian mixture model of the motion; (c) The desired motion the robot is used as its control input to perform the pouring with visual servoing.

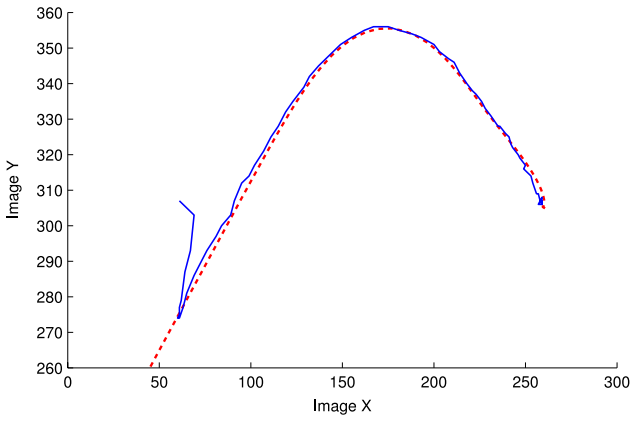

Fig. 14. The desired teapot trajectory in the image is shown in red dashed line; the real teapot trajectory (solid blue curve) in the image of the same camera shows the control result with the visual servoing by the FANUC robotic arm.

Fig. 14 shows one example of the robot's manipulation motion controlled by our visual servo controller, which follows the desired trajectory generated by GMR accurately. The robot was able to follow all the learned desired motion with our visual servoing approach.

5. Conclusions and future work

In this paper, we investigated object categorization and action recognition using an object–object-interaction affordance framework. The knowledge of object affordance is learned from labeled video sequences and represented as a Bayesian Network. The elements of the Bayesian Network include objects, human action, and object reaction. Experiments showed that with object–object-interaction affordance knowledge, the object classification rate, and especially the action recognition rate were significantly improved.

The learned affordance knowledge and interactive motions can be further used to teach robots manipulation skills that are associated with the paired objects. This paper demonstrated an image-based visual servoing approach to use the learned motion features of the affordance as the control goals to control a robotic arm to perform a manipulation task. The presented approach can also be integrated into other robotic systems to handle other manipulation tasks involving paired objects, and we plan to combine it with our learning from demonstration approach [39] in the future. The motion can be further analyzed and represented with more abstract forms such as motion grammar [40], and then integrated into existing skill learning techniques such as skill trees [1] to manage more complicated motions.

This work is currently limited to 2D motion from a 2D camera. It can be easily extended to the 3D space with either marker-based or range-sensor-based motion tracking. Another limitation is that this current Bayesian Network model is designed to process paired-objects. Even though paired-objects are very common in our daily living environment, not all relationships among objects are covered to refine our approaches and corroborate the usability of the learned affordance. In the future, we plan to model more complicated relationships among multiple objects. One direct expansion of this paper is to combine multiple Bayesian networks in a piecewise manner into a much bigger network if there are inter-connections between different pairs of objects.

Recognizing objects from a clustered scene is a broad and very active reach topic. Our work does not directly address its challenging problems, such as partial occlusion. Therefore, our work will not have a direct impact on clustered scene understanding. However, our approach can increase the object recognition success rate by including interaction affordance information-comparing without interaction affordance in the same setup. Therefore, our approach can be integrated into most state-of-the-art research work handling clustered scenes to increase the object recognition success rate.

References

[1] G.D. Konidaris, S. Kuindersma, R. Grupen, A. Barto, Robot learning from demonstration by constructing skill trees, Int. J. Robot. Res. 31 (3) (2012) 360–375.
[2] B.D. Argall, S. Chernova, et al., A survey of robot learning from demonstration, Robot. Auton. Syst. 57 (5) (2009) 469–483.
[3] S. Schaal, S. Ijspeert, et al., Computational approaches to motor learning by imitation, Philos. Trans. R. Soc. Lond. Ser. A Biol. Sci. 358 (1431) (2003) 537–547.
[4] A. Gupta, L. Davis, Objects in action: An approach for combining action understanding and object perception, in: Conference on Computer Vision and Pattern Recognition, 2007, pp. 1–8.
[5] H. Kjellstrom, J. Romero, D. Kragic, Visual object–action recognition: inferring object affordances from human demonstration, Comput. Vis. Image Underst. 115 (2010) 81–90.
[6] J. Gall, A. Fossati, L. Gool, Functional categorization of objects using real-time markerless motion capture, in: Conference on Computer Vision and Pattern Recognition, 2011, pp. 1969–1976.
[7] Y. Jiang, M. Lim, C. Zheng, A. Saxena, Learning to place new objects in a scene, Int. J. Robot. Res. 31 (9) (2012) 1021–1043.
[8] B. Yao, L. Fei-Fei, Modeling mutual context of object and human pose in human object interaction activities, in: Conference on Computer Vision and Pattern Recognition, 2010, pp. 17–24.

[9] M. Stark, P. Lies, M. Zillich, J. Wyatt, B. Schiele, Functional object class detection based on learned affordance cues, Comput. Vis. Syst. 5008 (2008) 435–444.
[10] H. Grabner, J. Gall, L. Van Gool, What makes a chair a chair? in: Conference on Computer Vision and Pattern Recognition, 2011, pp. 1529–1536.
[11] E. Aksoy, A. Abramov, F. Worgotter, B. Dellen, Categorizing object–action relations from semantic scene graphs, in: IEEE Intl. Conference on Robotics and Automation, 2010, pp. 398–405.
[12] J. Gibson, The theory of affordances, in: R. Shaw, J. Bransford (Eds.), Perceiving, Acting and Knowing, Erlbaum, Hillsdale, NJ, 1977.
[13] G. Rizzolatti, L. Craighero, The mirror neuron system, Ann. Rev. Neurosci. 27 (2004) 169–192.
[14] G. Rizzolatti, L. Craighero, Mirror neuron: a neurological approach to empathy, in: J.-P. Changeux, A.R. Damasio, W. Singer, Y. Christen (Eds.), Neurobiology of Human Values, Springer, Berlin and Heidelberg, 2005.
[15] E. Oztop, M. Kawato, M. Arbib, Mirror neurons and imitation: a computationally guided review, Epub Neural Netw. 19 (2006) 254–271.
[16] G. Di Pellegrino, L. Fadiga, L. Fogassi, V. Gallese, G. Rizzolatti, Understanding motor events: a neurophysiological study, Exp. Brain Res. 91 (1992) 176–180.
[17] V. Gallese, L. Fogassi, L. Fadiga, G. Rizzolatti, Action representation and the inferior parietal lobule, in: W. Prinz, B. Hommel (Eds.), Attention and Performance XIX. Common Mechanisms in Perception and Action, Oxford University Press, Oxford, 2002.
[18] E. Yoon, W. Humphreys, M. Riddoch, The paired-object affordance effect, J. Exp. Psychol. Human 36 (2010) 812–824.
[19] A. Borghi, A. Flumini, N. Natraj, L. Wheaton, One hand, two objects: emergence of affordance in contexts, Brain Cogn. 80 (1) (2012) 64–73.
[20] S. Thill, D. Caligiore, A. Borghi, T. Ziemke, G. Baldassarre, Theories and computational models of affordance and mirror systems: an integrative review, Neurosci. Biobehav. Rev. 37 (3) (2013) 491–521.
[21] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: Conference on Computer Vision and Pattern Recognition, 2005, pp. 886–893.
[22] P.F. Felzenszwalb, D. McAllester, D. Ramanan, A discriminatively trained, multiscale, deformable part model, in: Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8.
[23] J. Deng, W. Dong, R. Socher, L. Li, K. Li, L. Fei-Fei, Imagenet: A large-scale hierarchical image database, in: Int. Conf. on Computer Vision and Pattern Recognition, 2009, pp. 248–255.
[24] C. Chang, C. Lin, Libsvm: a library for support vector machines, ACM Trans. Intell. Syst. Technol. 2 (3) (2011) 1–27.
[25] M. Iacoboni, I. Molnar-Szakacs, V. Gallese, G. Buccino, J. Mazziotta, et al. Grasping the intentions of others with one's own mirror neuron system, PLoS Biol. 3 (3).
[26] S.A. Jax, L.J. Buxbaum, Response interference between functional and structural actions linked to the same familiar object, Cognition 115 (2) (2010) 350–355.
[27] A. Argyros, M. Lourakis, Real-time tracking of multiple skin-colored objects with a possibly moving camera, in: European Conference on Computer Vision, 2004, pp. 368–379.
[28] C. Conaire, N.E. O'Connor, A.F. Smeaton, Detector adaptation by maximising agreement between independent data sources, in: IEEE International Workshop on Object Tracking and Classification Beyond the Visible Spectrum, 2007, pp. 1–6.
[29] Z. Kala, J. Matas, K. Mikolajczyk, P–n learning: Bootstrapping binary classifiers by structural constraints, in: Conference on Computer Vision and Pattern Recognition, 2010, pp. 49–56.
[30] P. Buehler, M. Everingham, D.P. Huttenlocher, A. Zisserman, Upper body detection and tracking in extended signing sequences, Int. J. Comput. Vis. 95 (2011) 180–197.
[31] V. Prasad, V. Kellokompu, L. Davis, Ballistic hand movements, in: F. Perales, R. Fisher (Eds.), Articulated Motion and Deformable Objects, 2006, pp. 153–164.
[32] L. Fogassi, et al., Parietal lobe: from action organization to intention understanding, Science 29 (308) (2005) 662–667.
[33] A. Hamilton, S. Grafton, The motor hierarchy: from kinematics to goals and intentions, in: P. Haggard, Y. Rosetti, M. Kawato (Eds.), Attention and Performance, 2007, Ch. 22.
[34] J. Pearl, Probabilistic Reasoning in Intelligent Systems: Network and Plausible Inference, Morgan Kaufmann, 1988.
[35] S. Hutchinson, G. Hager, P. Corke, A tutorial on visual servo control, IEEE Trans. Robot. Automat. 12 (5) (1996) 651–670.
[36] F. Chaumette, S. Hutchinson, Visual servo control. I. Basic approaches, IEEE Robot. Autom. Mag. 13 (4) (2006) 82–90.
[37] W.G. Pence, F. Farelo, A.R.M. Alqasemi, Y. Sun, R. Dubey, Visual servoing control of a 9-dof wmra to perform adl tasks, in: IEEE Intl. Conference on Robotics and Automation, 2012, pp. 916–922.
[38] S. Calinon, A. Billard, Incremental learning of gestures by imitation in a humanoid robot, in: Proceedings of the ACM/IEEE International Conference on Human–Robot Interaction, 2007, pp. 255–262.
[39] Y. Lin, S. Ren, M. Clevenger, Y. Sun, Learning grasping force from demonstration, in: IEEE Intl. Conference on Robotics and Automation, 2012, pp. 1526–1531.
[40] N. Dantam, M. Stilman, The motion grammar: Linguistic perception, planning, and control, in: Proceedings of Robotics: Science and Systems, 2011, pp. 1–8.

Yu Sun is currently an Assistant Professor in the Department of Computer Science and Engineering at the University of South Florida. He received his Ph.D. degree in Computer Science from the University of Utah, Salt Lake City, in 2007. He was a Postdoctoral Associate at Mitsubishi Electric Research Laboratories (MERL), Cambridge, M.A. from Dec. 2007 to May 2008 and a Postdoctoral Associate in the School of Computing at the University of Utah from May 2008 to May 2009. He received his B.S. and M.S. degrees in Electrical Engineering from Dalian University of Technology, Dalian, China, in 1997 and 2000, respectively. He currently serves as an Associate Editor of the IEEE Robotics & Automation Magazine. His research interests include medical imaging, robotics, haptics, computer vision, virtual reality, human computer interaction (HCI), and medical applications.

Shaogang Ren received his B.E. degree from Central South University in 2006, and M.E. degree from Huazhong University of Science and Technology in 2008. Since 2010 he has been a Ph.D. student in the Computer Science and Engineering Department, University of South Florida. His research interests include machine learning and computer vision.

Yun Lin received her B.E. and M.E. degrees from the University of Science and Technology Beijing in 2005 and 2008, respectively. Since 2009, she has been a Ph.D. student in the Computer Science and Engineering Department, University of South Florida. Her research interests include robotics, grasping, manipulation, and haptics.