Visuospatial Skill Learning for Object Reconfiguration Tasks

Seyed Reza Ahmadzadeh,   Petar Kormushev,   Darwin G. Caldwell

***Abstract*— We present a novel robot learning approach based on visual perception that allows a robot to acquire new skills by observing a demonstration from a tutor. Unlike most existing learning from demonstration approaches, where the focus is placed on the trajectories, in our approach the focus is on achieving a desired goal configuration of objects relative to one another. Our approach is based on visual perception which captures the object's context for each demonstrated action. This context is the basis of the visuospatial representation and encodes implicitly the relative positioning of the object with respect to multiple other objects simultaneously. The proposed approach is capable of learning and generalizing multi-operation skills from a single demonstration, while requiring minimum *a priori* knowledge about the environment. The learned skills comprise a sequence of operations that aim to achieve the desired goal configuration using the given objects. We illustrate the capabilities of our approach using three object reconfiguration tasks with a Barrett WAM robot.**

I. Introduction

Several robot skill learning approaches based on human demonstrations have been proposed during the past years. Many of them address motor skill learning in which new motor skills are transferred to the robot using policy derivation techniques (e.g. mapping function [1], system model [2]).

Motor skill learning approaches can be categorized in two main subsets: trajectory-based and goal-based approaches. Trajectory-based approaches put the focus on recording and re-generating trajectories for object manipulation [3], [4]. However, in many cases, it is not the trajectory that is important but the goal of the action, for example, solving a jigsaw puzzle [5] or assembling an electric circuit board. In such examples, trajectory-based approaches actually increase the complexity of the learning process unnecessarily.

Several goal-based approaches such as [6] have been developed to address this issue. For instance, there is a large body of literature on grammars from the linguistic and computer science communities, with a number of applications related to robotics [7], [8]. Other symbolic learning approaches are focused on goal configurations rather than action execution [9]. Such approaches inherently comprise many steps, for instance, segmentation, clustering, object recognition, structure recognition, symbol generation, syntactic task modeling, motion grammar and rule generation, etc. Another drawback of such approaches is that they require a significant amount of *a priori* knowledge to be manually engineered into the system. Furthermore, most above-mentioned approaches assume the availability of the information on the internal state of a demonstrator such as joint angles, while humans usually cannot directly access to imitate the observed behavior.

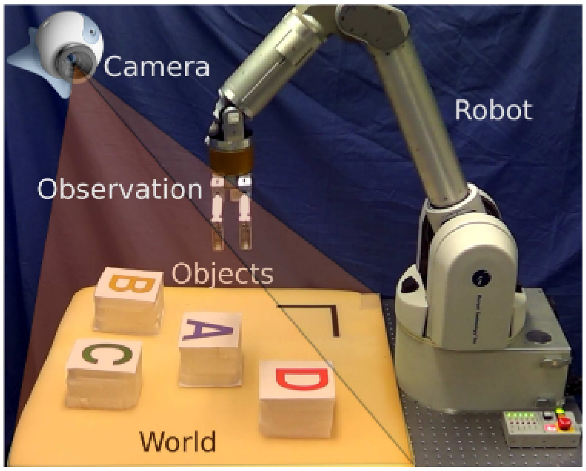


Fig. 1. The experimental setup for a visuospatial skill learning (VSL) task.

An Alternative to motor skill learning approaches are visual skill learning approaches [10], [11]. These approaches are based on observing the human demonstration and using human-like visuospatial skills to replicate the task. Visuospatial skill is the capability to visually perceive the spatial relationship between objects.

In this paper, we propose a novel visuospatial skill learning approach for robot object reconfiguration tasks. Unlike the motor skill learning approaches, our approach utilizes visual perception as the main information source for learning object reconfiguration skills from demonstration. The proposed visuospatial skill learning (VSL) approach uses a simple algorithm and minimum *a priori* knowledge to learn a sequence of operations from a single demonstration. In contrast to many previous approaches, VSL leverages simplicity, efficiency, and user-friendly human-robot interaction. Rather than relying on complicated models of human actions, labeled human data, or object recognition, our approach allows the robot to learn a variety of complex tasks effortlessly, simply by observing and reproducing the visual relationship among objects. We demonstrate the feasibility of the proposed approach in three real-world experiments in which the robot learns to organize objects of different shape and color on a tabletop workspace to accomplish a goal configuration.

This research was partially supported by the PANDORA EU FP7 project under the grant agreement No. ICT-288273. Project website: http://persistentautonomy.com/.

Authors are with the Department of Advanced Robotics, Istituto Italiano di Tecnologia, via Morego, 30, 16163 Genova, Italy. {reza.ahmadzadeh, petar.kormushev, darwin.caldwell} at iit.it

II. Related Work

Visual skill learning or learning by watching is one of the most powerful mechanisms of learning in humans. Researchers have shown that even newborns can imitate simple body movements such as facial gestures [12]. In cognitive science, learning by watching has been investigated as a source of higher order intelligence and fast acquisition of knowledge [10], [13]. In the rest of this section we give examples for visual skill learning approaches.

In [10], a robot agent watches a human teacher performing a simple assembly task in a tabletop environment. The motor movements of the human are classified as actions known to the robot (e.g. pick, move, place etc.). The robot can reproduce the sequence of actions even if the initial configuration of the objects are changed. In their approach, the set of actions is already known to the robot and they use symbolic labels to reproduce the sequence of actions. In order to detect the movement of an object, they detect and track the demonstrator's hand. Since they rely solely on passive observations of a demonstration, this method has to make use of complex computer vision techniques, in carefully structured environments, in order to infer all the information necessary for the task. In our method we neither use symbolic labels nor track the tutor's hand. Asada et al. [14] propose a method for learning by observation (teaching by showing) based on the demonstrator's view recovery and adaptive visual servoing. They believe that coordinate transformation is a time-consuming and error-prone method. Instead, they assume that both the robot and the demonstrator have the same body structure. They use two sets of stereo cameras, one for observing the robot's motion and the other for observing the demonstrator's motion. The optic-geometrical constraint, called ”epipolar constraint”, is used to reconstruct the view of the agent, on which adaptive visual servoing is applied to imitate the observed motion. In our method, we use coordinate transformation.

In [15], the authors propose an approach using multiple sensors in a kitchen environment with typical household tasks. They focus on pick-and-place operations including techniques for grasping. They extract finger joint movements and hand position in 3D space from a data glove. In addition, a magnetic field-based tracking system and an active trinocular camera head was used for object recognition using vision approaches. The method is based on pre-trained neural networks to detect hand configurations and to search in a predefined symbol database. However, there was no real-world reproduction with a robot. A similar research focuses on extracting and classifying subtasks for grasping tasks using visual data from demonstration, generating trajectory and extracting subtasks [16]. They use color markers to capture data from the tutor's hand. In our method, we use neither neural networks nor symbol abstraction techniques. A visual learning by Imitation approach is presented in [11]. The authors utilize neural networks to map visual perception to motor skills (visuo-motor) together with viewpoint transformation. For gesture imitation a Bayesian formulation is adopted. A mono camera was used in their experiments. A method for segmenting demonstrations, recognizing repeated skills, and generalizing complex tasks from unstructured demonstration is presented in [7]. The authors use Beta Process Autoregressive HMM for recognizing and generalizing. They apply simple metrics on the captured object's context to distinguish between observations. They use pre-defined coordinated frames and visual fiducial fixed on each object for object detection. Furthermore, they clustered points in each frame and created a DMP for each segment in the demonstration. In our method we use captured object's context to find pick-and-place points for each operation instead of object recognition and point clustering.Visual analysis of demonstrations and automatic policy extraction for an assembly task is presented in [8]. To convert a demonstration of the desired task into a string of connected events, this approach uses a set of different techniques such as image segmentation, clustering, object recognition, object tracking, structure recognition, symbol generation, transformation of symbolic abstraction, and trajectory generation. In our approach we do not use symbol generation techniques.

A robot goal learning approach is presented in [9] that can ground discrete concepts from continuous perceptual data using unsupervised learning. The authors provided the demonstrations of five pick-and-place tasks in a tabletop workspace by non-expert users. They utilize object detection, segmentation, and localization methods using color markers. During each operation, the approach detects the object that changes most significantly. They provide the starting and ending time of each demonstration using graphical or speech commands. Our approach solely relies on the captured observations to learn the sequence of operations and there is no need to perform any of those steps. In [17] a database of features (all possible configurations of the objects) is created and marked by supervised learning. A beta regression classifier was used to learn the features to detect ‘good’ and ‘bad’ configurations. Since the system is relied on a large bank of features, they discretized state space and used methods like simulated annealing and gradient approaches to find an optimal solution. They used a robot to demonstrate room tidying task. our approach, on the other hand, does not depend on a huge bank of features.

III. Visuospatial Skill Learning

In this section, we introduce the visuospatial skill learning (VSL) approach. After stating our assumptions and defining the basic terms, we describe the problem statement and explain the VSL algorithm.

In this paper, object reconfiguration tasks consisting of pick-and-place actions are considered in which achieving the goal of the task and retaining the sequence of operations are particularly important. As can be seen in Fig. 1 the experimental set-up for all the conducted experiments consists of a torque-controlled 7-DOF Barrett WAM robotic arm with 3-finger Barrett Hand attached, a tabletop working area, a set of objects, and a CCD camera which is mounted above the workspace.

A tutor demonstrates a sequence of operations on the available objects. Each operation consists of one pick action and one place action which are captured by the camera. Afterwards, using our proposed method, and starting from a random initial configuration of the objects, the manipulator can perform a sequence of new operations which ultimately results in reaching the same goal as the one demonstrated by the tutor.

A. *Terminology*

To describe our learning approach accurately, we need to define some basic terms.

***World*:** In our method, the intersection area between the robot's workspace and the demonstrator's workspace which is observable by the camera (with a specific field of view) is called a *world*. The *world* includes objects which are being used during the learning task, and can be reconfigured by the human tutor and the robot.

***Frame*:** A bounding box which defines a cuboid in 3D or a rectangle in 2D coordinate system. The size of the *frame* can be fixed or variable. The maximum size of the *frame* is equal to the size of the *world*.

***Observation*:** The captured context of the *world* from a predefined viewpoint and using a specific *frame*. Each *observation* is a still image which holds the content of that part of the *world* which is located inside the *frame*.

***Pre-pick observation*:** An *observation* which is captured just before the pick action and is centered around the pick location.

***Pre-place observation*:** An *observation* which is captured just before the place action and is centered around the place location.

The *pre-pick* and *pre-place* terms in our approach are analogous to the *precondition* and *postcondition* terms used in logic programming languages (e.g. Prolog).

B. *Problem Statement*

Formally, we define a process of visuospatial skill learning as a tuple $\mathcal{V} = \{\eta, \mathcal{W}, \mathcal{O}, \mathcal{F}, \mathcal{S}\}$ where, η defines the number of operations which can be specified in advance or can be detected during the demonstration phase; $\mathcal{W} \in \Re^{m \times n}$ is a matrix containing the image which represents the context of the *world* including the workspace and all objects; $\mathcal{O}$ is a set of *observation* dictionaries $\mathcal{O} = \{\mathcal{O}_{Pick}, \mathcal{O}_{Place}\}$; $\mathcal{O}_{Pick}$ is an *observation* dictionary comprising a sequence of *pre-pick observations* $\mathcal{O}_{Pick} = \langle \mathcal{O}_{Pick}(1), \mathcal{O}_{Pick}(2), \ldots, \mathcal{O}_{Pick}(\eta) \rangle$, and $\mathcal{O}_{Place}$ is an *observation* dictionary comprising a sequence of *pre-place observations* $\mathcal{O}_{Place} = \langle \mathcal{O}_{Place}(1), \mathcal{O}_{Place}(2), \ldots, \mathcal{O}_{Place}(\eta) \rangle$. For example, $\mathcal{O}_{Pick}(i)$ represents the *pre-pick observation* captured during the i^{th} operation. $\mathcal{F} \in \Re^{m \times n}$ is an *observation frame* which is used for recording the *observations*; and $\mathcal{S}$ is a vector which stores scalar scores related to each detected match during a match finding process. The score vector is calculated using a metric by comparing the new observations captured during the reproduction phase with the dictionary of recorded observations in the demonstration phase.

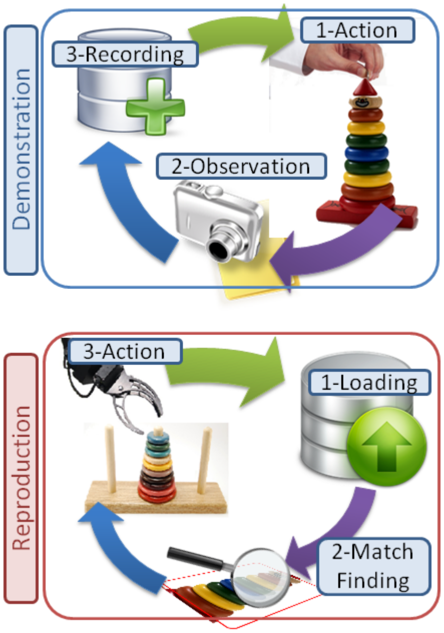


Fig. 2. A high-level flow diagram illustrating the demonstration and reproduction phases in the VSL approach.

The output of the reproduction phase is the set $\mathcal{P} = \{\mathcal{P}_{Pick}, \mathcal{P}_{Place}\}$ which contains the identified positions for performing the pick-and-place operations by the VSL algorithm. $\mathcal{P}_{Pick}$ is an ordered set of *pre-pick* positions $\mathcal{P}_{Pick} = \langle \mathcal{P}_{Pick}(1), \mathcal{P}_{Pick}(2), \ldots, \mathcal{P}_{Pick}(\eta) \rangle$. $\mathcal{P}_{Place}$ is an ordered set of *pre-place* positions $\mathcal{P}_{Place} = \langle \mathcal{P}_{Place}(1), \mathcal{P}_{Place}(2), ..., \mathcal{P}_{Place}(\eta) \rangle$. For example, $\mathcal{P}_{Pick}(i)$ represents the pick-position during the i^{th} operation.

C. *Methodology*

The VSL approach consists of two phases: demonstration and reproduction. A high-level flow diagram illustrating the VSL approach is shown in Fig. 2. In both phases, initially, the objects are randomly placed in the *world*, $\mathcal{W} = \{\mathcal{W}_D, \mathcal{W}_R\}$. (So, VSL does not depend on initial configuration of the objects.) $\mathcal{W}_D$ and $\mathcal{W}_R$ represent the *world* during the demonstration and the reproduction phases respectively.

Pseudo-code of the proposed implementation of VSL is given in Algorithm 1. The initial step in each learning task is calculating the (2D to 2D) coordinate transformation (line 1 in Algorithm 1) which means computing a homography matrix, $\mathcal{H} \in \Re^{3 \times 3}$ to transform points from the image plane to the workspace plane (Fig. 3). In the demonstration phase, the tutor starts to reconfigure the objects to fulfill a desired goal which is not known to the robot. During this phase the size of the *observation frame*, $\mathcal{F}_D$, is fixed and can be equal or smaller than the size of the *world*, $\mathcal{W}_D$. Two functions, *RecordPrePickObs* and *RecordPrePlaceObs*, are developed to capture and rectify one *pre-pick observation* and one *pre-place observation* for each operation. The captured images are rectified using the calculated homography matrix.

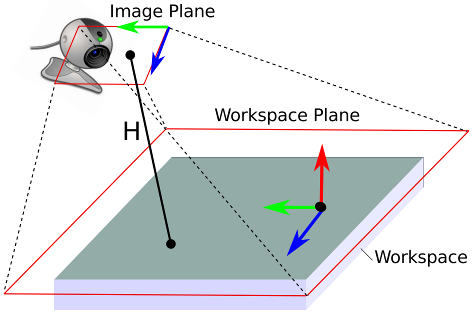


Fig. 3. Coordinate transformation from the image plane to the workspace plane (H is the homography matrix).

```
    Input  : $\{\eta, \mathcal{W}, \mathcal{F}\}$
    Output: $\{\mathcal{P}\}$
1   Initialization: $CalculateTransformation$
    // Part I : Demonstration
2   for $i = 1$ to $\eta$ do
3       $\mathcal{O}_{Pick}(i) = RecordPrePickObs(\mathcal{W}_D, \mathcal{F}_D)$
4       $\mathcal{O}_{Place}(i) = RecordPrePlaceObs(\mathcal{W}_D, \mathcal{F}_D)$
5   end
    // Part II : Reproduction
6   $Re-calculate\ Transformation(if\ necessary)$
7   for $i = 1$ to $\eta$ do
8       $\mathcal{F}^* = FindBestMatch(\mathcal{W}_R, \mathcal{O}_{Pick}(i))$
9       $\mathcal{P}_{Pick}(i) = FindPickPosition(\mathcal{F}^*)$
10      $PickObjectFromPosition(\mathcal{P}_{Pick}(i))$
11      $\mathcal{F}^* = FindBestMatch(\mathcal{W}_R, \mathcal{O}_{Place}(i))$
12      $\mathcal{P}_{Place}(i) = FindPlacePosition(\mathcal{F}^*)$
13      $PlaceObjectToPosition(\mathcal{P}_{Place}(i))$
14  end
```

Algorithm 1: Pseudo-code for the VSL (Visuospatial Skill Learning) approach.

Using image subtraction and thresholding techniques, the two mentioned functions (lines 3 and 4 in Algorithm 1) extract a pick point and a place point in the captured *observations* and then center the *observations* around the extracted positions. The recorded sets of *observations*, $\mathcal{O}_{Pick}$ and $\mathcal{O}_{Place}$, form an *observation* dictionary which is used as the input for the reproduction phase of VSL. Although each capturing process, during the demonstration, is initiated by the demonstrator, the whole procedure can be performed automatically using some image processing techniques like optical flow.

Before starting the reproduction phase, if the position and orientation of the camera with respect to the robot is changed we need to re-calculate the homography matrix for the new configuration of the coordinate systems. The next phase, reproduction, starts with randomizing the objects in the *world* to create a new *world*, $\mathcal{W}_R$. Then the algorithm selects the first *pre-pick observation* and searches for similar visual perception in the new *world* to find the best match (using the *FindBestMatch* function in lines 8 and 10). Although, the *FindBestMatch* function can use any metric to find the best matching observation, to be consistent, in all of our experiments we use the same metric (see section IV-B). This metric produces a scalar score with respect to each comparison. If more than one match is found, the *FindBestMatch* function returns the match with higher score. After the algorithm finds the best match, the *FindPickPosition* function (line 9) identifies the pick position which is located at the center of the corresponding *frame*, $\mathcal{F}^*$. The *PickObjectFromPosition* uses the result from the previous step to pick the object from the identified position. Then, the algorithm repeats the searching process to find the best match with the corresponding *pre-place observation* (line 11). The *FindPlacePosition* function identifies the place position at the center of the corresponding *frame*, $\mathcal{F}^*$ and the *PlaceObjectToPosition* function uses the result from the previous step to place the object to the identified position. The *frame* size in the reproduction phase is fixed and equal to the size of the *world* ($\mathcal{W}_R$). One of the advantages of the VSL approach is that it provides the exact position of the object where the object should be picked without using any object detection methods. The reason is, when the algorithm is generating the *observations*, it centers the *observation frame* around the point that the demonstrator is operating the object.

IV. Implementation of VSL

In this section we describe the steps required to implement the VSL approach for real-world experiments.

A. Calculating Coordinate Transformation (Homography)

In each experiment the camera's frame of reference may vary with respect to the robot's frame of reference. To transform points between these coordinate systems we calculate the homography. Using, at least, 4 real points on the workspace plane and 4 corresponding points on the image plane, the homography is calculated using singular value decomposition. The extracted homography matrix is used not only for coordinate transformation but also to rectify the raw captured images from the camera. For each task the process of extracting the homography should be repeated whenever the camera's frame of reference is changed with respect to the robot's frame of reference.

Using coordinate transformation, makes VSL approach view-invariant. It means, after the robot acquires a new skill, it still can reproduce the skill afterwards even if the experimental setup is changed. It just need to use the new coordinate transformation.

B. Image Processing

Image processing methods have been used in both demonstration and reproduction phases of VSL. In the demonstration phase, for each operation we capture a set of raw images consist of *pre-pick*, *pre-place* images. Firstly, we rectify the captured raw images using the homography matrix. Secondly we apply image subtraction and thresholding on the couple

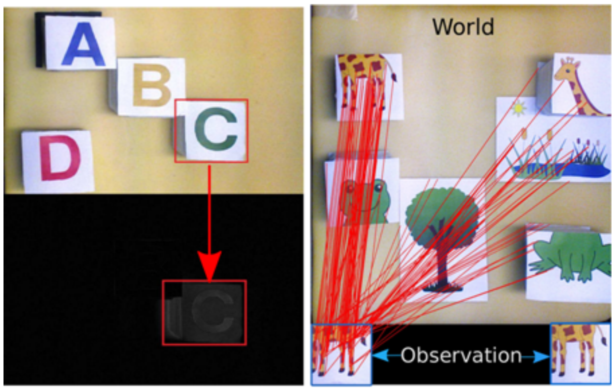


Fig. 4. The result of image subtracting and thresholding for a place action (left), Match finding result between the second observation and the world in the 2nd operation of the 'Animal Puzzle' task using SIFT (right).

of images to generate *pre-pick* and *pre-place observations*. The produced *observations* are centered on the *frame*. In the reproduction phase, for each operation we rectify the captured *world observation*. Then, we load the corresponding recorded *observations* from demonstration phase and apply a metric to find the best match (the *FindBestMatch* function in the Algorithm 1). Although any metric can be used in this function (e.g. window search method), we use Scale Invariant Feature Transform (SIFT) algorithm [18]. SIFT is one of the most popular feature-based methods which is able to detect and describe local features that are invariant to scaling and rotation. Afterwards, we apply RANSAC in order to estimate the transformation matrix from the set of matched points. Fig. 4 shows the result of the SIFT algorithm applying to an *observation* and a new *world*.

VSL relies on vision, which might be obstructed by other objects, by the demonstrator's body, or during reproduction by the robot's arm. Therefore, for physical implementation of the VSL approach special care needs to be taken to avoid such obstructions.

Finally, we should mention that the image processing part is not the focus of this paper, and we use the SIFT-RANSAC algorithms because of their popularity and the capability of fast and robust match finding.

C. Trajectory Generation

We use the pick and place points extracted from the image processing section to generate a trajectory for the corresponding operation. For each pick-and-place operation the desired Cartesian trajectory of the end-effector is a cyclic movement between three key points: rest point, pick point, and place point. Fig 5 illustrates two different views of a generated trajectory. The robot starts from the rest point (point no.1) and moves smoothly (along the red curve) towards the pick point (point no.2). Then the robot picks up an object, relocates it (along the green curve) to the place-point (point no.3), places the object there, and finally moves back (along the blue curve) to the rest point. For each part of the trajectory, including the grasping and releasing parts, we define a specific duration and initial boundary conditions (initial positions and velocities). We define a geometric path in workspace which can be expressed in the parametric form of the equations (1) to (3), where s is defined as a function of time t, ($s = s(t)$), and we define the different elements of the geometric path as $p_x = p_x(s)$,$p_y = p_y(s)$, and $p_z = p_z(s)$. Polynomial function of order three with initial condition of position and velocity is used to express p_x, and p_y. Also, We use equation (3) for p_z. Distributing the time smoothly with a 3rd order polynomial starting from initial time to final time, together with the equations (1)-(3), the generated trajectories are suitable for obstacle avoidance while picking or placing objects besides each others.

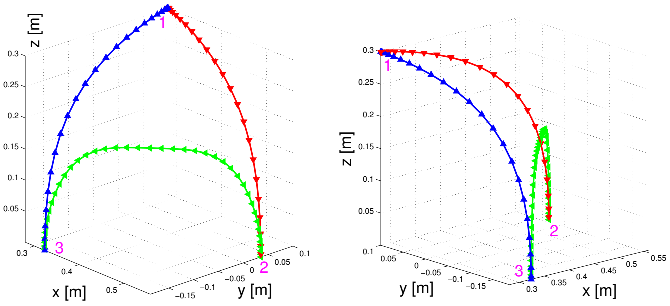


Fig. 5. The generated trajectory for the first pick-and-place operation of the ABCD task from two viewpoints. Point 1: rest-point, Point 2: pick-point, and Point 3: place-point.

TABLE I

CAPABILITIES OF VSL ILLUSTRATED IN EACH TASK

Task	Animal Puzzle	Alphabet Ordering	Tower of Hanoi
Relative Positioning	✓	✓	✓
Absolute Positioning	-	✓	✓
User intervention to modify the reproduction	-	-	✓
Multiple operations performed on the same object	-	-	✓

$$p_x = a_3 s^3 + a_2 s^2 + a_1 s + a_0 \tag{1}$$

$$p_y = b_3 s^3 + b_2 s^2 + b_1 s + b_0 \tag{2}$$

$$p_z = h[1 - |(\tanh^{-1}(h_0(s - 0.5)))^{\kappa}|] \tag{3}$$

V. Experimental Results

To illustrate the capabilities and the limitations of our approach, three real-world experiments are demonstrated in this section. As summarized in Table I, the proposed experiments show four different capabilities of VSL, including absolute and relative positioning, user intervention to modify the reproduction, and multiple operations performed on the same object. The video accompanying this paper shows the execution of the tasks and is available online at [19]. In all of the experiments, in the demonstration phase, we set the size of the *frame* for the *pre-pick observation* equal to the size of the biggest object in the *world*, and the size of the *frame* for the *pre-place observation* 2 to 3 times bigger than the size of the biggest objects in the *world*. In the reproduction

phase, on the other hand, we set the size of the *frame* equal to the size of the *world*.
The camera is mounted above the table facing the workspace. The resolution of the captured images are 1280×960 pixels. Although the trajectory is created in the end-effector space, we control the robot in the joint-space based on the inverse dynamics to avoid singularities. Also, during the reproduction phase, our controller keeps the orientation of the robot's hand (end-effector) perpendicular to the workspace, in order to faciliate the pick-and-place operation.
Table II lists the execution time for different steps of the implementation of the VSL approach.

A. Task I - Alphabet Ordering

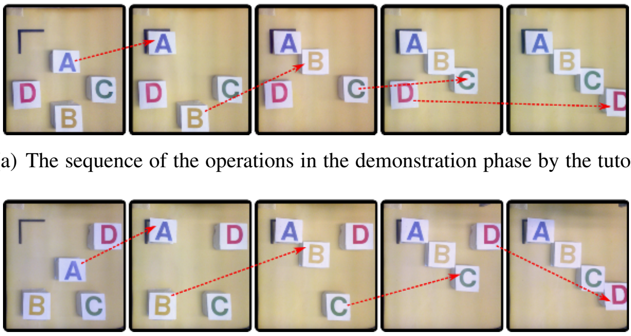
In the first VSL task, the *world* includes four cubic objects labeled with A, B, C, and D letters and a fixed right angle baseline which is a static part of the *world*. The goal is to reconfigure the set of objects with respect to the baseline according to the demonstration. This task emphasizes VSL's capability of relative positioning of an object with respect to other surrounding objects in the *world* (a visuospatial skill). This inherent capability of VSL is achieved through the use of visual *observations* which capture both the object of interest and its surrounding objects (i.e. its context). In addition, the baseline is provided to show the capability of absolute positioning of the VSL approach. It shows the fact that we can teach the robot to attain absolute positioning of objects without defining any explicit *a priori* knowledge. Fig. 6(a) shows the sequence of the operations in the demonstration phase. Recording *pre-pick* and *pre-place observations*, the robot learns the sequence of operations. Fig. 6(b) shows the sequence of the operations produced by VSL starting from a novel *world* (i.e. new initial configuration) which is achieved by randomizing the objects in the *world*.

B. Task II - Animal Puzzle

In the previous task, duo to the absolute positioning capability of VSL, the final configuration of the objects in the reproduction and the demonstration phases are always the same. In this experiment, however, the final result can be a totally new configuration of objects by removing the fixed baseline from the *world*. The goal of this experiment is to show the VSL's capability of relative positioning. In this VSL task, the *world* includes two sets of objects which complete a 'frog' puzzle beside a 'pond' label together with a 'giraffe' puzzle beside a 'tree' label. Fig. 7(a) shows the sequence of the operations in the demonstration phase. To show the capability of generalization, the 'tree' and the 'pond' labels are randomly replaced by the tutor before the reproduction phase. The goal is to assemble the set of objects for each animal with respect to the labels according to the demonstration. Fig. 7(b) shows the sequence of the operations reproduced by VSL.

TABLE II
EXECUTION TIME FOR DIFFERENT STEPS OF VSL

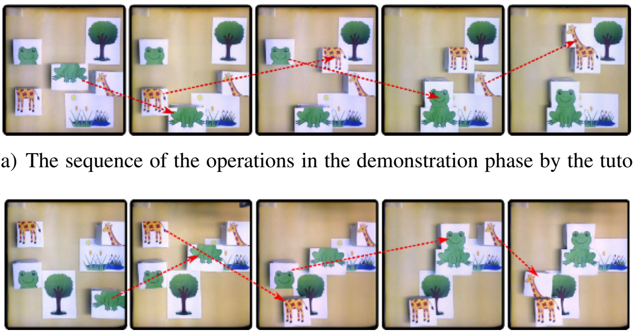
Steps	**Time (sec)**	For each
Homography	0.2-0.35	learning task
SIFT+RANSAC	18-26	operation (reproduction)
Image subtraction	0.09-0.11	comparison
Image thresholding	0.13-0.16	comparison
Trajectory generating	1.3-1.9	operation (reproduction)
Trajectory tracking	6.6-15	operation (reproduction)
Image rectifying	0.3-0.6	image
Demonstration (calculation time)	3.8-5.6	operation
Reproduction	27.5-44.1	operation



(a) The sequence of the operations in the demonstration phase by the tutor

(b) The sequence of the operations in the reproduction phase by the robot

Fig. 6. Alphabet ordering (Details in Section V-A). The initial configuration of the objects in the *world* is different in (a) and (b). The red arrows show the operations.


(a) The sequence of the operations in the demonstration phase by the tutor

(b) The sequence of the operations in the reproduction phase by the robot

Fig. 7. Animal puzzle (Details in Section V-B). The initial and the final configurations of the objects in the *world* are different in (a) and (b). The red arrows show the operations.

C. Task III - Tower of Hanoi

The last experiment is the famous mathematical puzzle, Tower of Hanoi, consisting of a number of disks of different sizes and 3 bases or rods. The objective of the puzzle is to move the entire stack to another rod. This experiment demonstrates almost all capabilities of the VSL approach. Two of these capabilities are not accompanied by the previous experiments. Firstly, our approach enables the user to intervene to modify the reproduction. so, the robot can move the Hanoi disks to another base. (e.g. to move the stack of disks to the third base, instead of the second.) This goal can be achieved only if the user performs the very first operation in the reproduction phase and moves the smallest disk on the third base instead of the second.

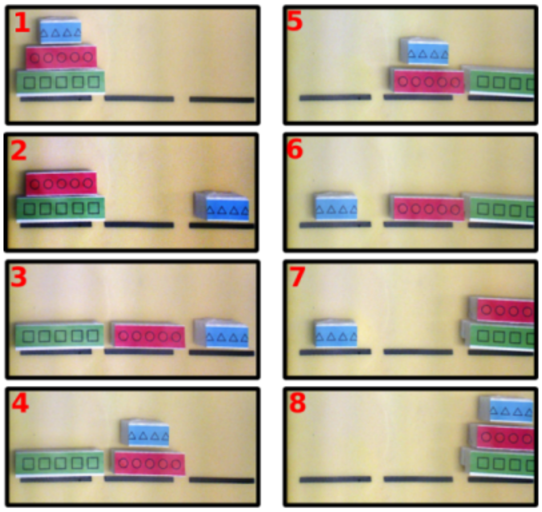


Fig. 8. The sequence of the reproduction for the Tower of Hanoi experiment to illustrate the main capabilities of VSL. (details in section V-C).

Secondly, the VSL approach enables the user to perform multiple operations on the same object during the learning task. As shown in Table I, this task also illustrates other capabilities of the VSL approach including the relative and absolute positioning. In Fig. 8 due to the lack of space we just provided the sequence of the reproduction.

VI. Discussion

In order to test the repeatability of our approach and to identify the possible factors of failure, we used the captured images from the real-world experiments while excluding the robot from the loop. We kept all other parts of the loop intact and repeated each experiment three times. The result shows that 6 out of 45 pick-and-place operations failed. The failure factors can be listed as: match finding error, noise in the images, unadjusted thresholding gain, and occlusion of the objects. Despite the fact that the algorithm is scale-invariant, color-invariant, and view-invariant, it has some limitations. For instance, if the tutor accidentally moves one object while operating another, the algorithm may fail to find a pick/place point. One possible solution is to use classification techniques together with the image subtraction and thresholding techniques to detect multi-object movements.

VII. CONCLUSION AND FUTURE WORK

We proposed a visuospatial skill learning approach that has powerful capabilities as shown in the three presented tasks. The method possesses the following capabilities: relative and absolute positioning, user intervention to modify the reproduction, and multiple operations performed on the same object. These characteristics make VSL a suitable choice for goal-based object reconfiguration tasks which rely on visual perception. Moreover, our approach is suitable for the vision-based robotic platforms which are designed to perform a variety of repetitive production tasks (e.g. Baxter). The reason is that applying VSL to such robots, requires no complex programming skill or costly integration.

ACKNOWLEDGMENT

The authors would like to thank Dr. Enver Sangineto for his invaluable suggestions about the image processing part.

References

[1] S. Vijayakumar and S. Schaal, "Locally weighted projection regression: An o (n) algorithm for incremental real time learning in high dimensional space," in *Proceedings of the Seventeenth International Conference on Machine Learning (ICML 2000)*, vol. 1, 2000, pp. 288–293.

[2] P. Abbeel and A. Ng, "Apprenticeship learning via inverse reinforcement learning," in *Proceedings of the twenty-first international conference on Machine learning*. ACM, 2004, p. 1.

[3] A. Ijspeert, J. Nakanishi, and S. Schaal, "Learning attractor landscapes for learning motor primitives," *Advances in neural information processing systems*, vol. 15, pp. 1523–1530, 2002.

[4] S. Calinon, F. Guenter, and A. Billard, "On learning, representing, and generalizing a task in a humanoid robot," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 37, no. 2, pp. 286–298, 2007.

[5] B. Burdea and H. Wolfson, "Solving jigsaw puzzles by a robot," *Robotics and Automation, IEEE Transactions on*, vol. 5, no. 6, pp. 752–764, 1989.

[6] D. Verma and R. Rao, "Goal-based imitation as probabilistic inference over graphical models," *Advances in neural information processing systems*, vol. 18, p. 1393, 2006.

[7] S. Niekum, S. Osentoski, G. Konidaris, and A. Barto, "Learning and generalization of complex tasks from unstructured demonstrations," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012.

[8] N. Dantam, I. Essa, and M. Stilman, "Linguistic transfer of human assembly tasks to robots," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012.

[9] C. Chao, M. Cakmak, and A. Thomaz, "Towards grounding concepts for transfer in goal learning from demonstration," in *Development and Learning (ICDL), 2011 IEEE International Conference on*, vol. 2. IEEE, 2011, pp. 1–6.

[10] Y. Kuniyoshi, M. Inaba, and H. Inoue, "Learning by watching: Extracting reusable task knowledge from visual observation of human performance," *Robotics and Automation, IEEE Transactions on*, vol. 10, no. 6, pp. 799–822, 1994.

[11] M. Lopes and J. Santos-Victor, "Visual learning by imitation with motor representations," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 35, no. 3, pp. 438–449, 2005.

[12] A. Meltzoff and M. Moore, "Newborn infants imitate adult facial gestures," *Child development*, pp. 702–709, 1983.

[13] G. Rizzolatti, L. Fadiga, V. Gallese, and L. Fogassi, "Premotor cortex and the recognition of motor actions," *Cognitive brain research*, vol. 3, no. 2, pp. 131–141, 1996.

[14] M. Asada, Y. Yoshikawa, and K. Hosoda, "Learning by observation without three-dimensional reconstruction," *Intelligent Autonomous Systems (IAS-6)*, pp. 555–560, 2000.

[15] M. Ehrenmann, O. Rogalla, R. Zöllner, and R. Dillmann, "Teaching service robots complex tasks: Programming by demonstration for workshop and household environments," in *Proceedings of the 2001 International Conference on Field and Service Robots (FSR)*, vol. 1, 2001, pp. 397–402.

[16] M. Yeasin and S. Chaudhuri, "Toward automatic robot programming: learning human skill from visual data," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 30, no. 1, pp. 180–185, 2000.

[17] M. Mason and M. Lopes, "Robot self-initiative and personalization by learning through repeated interactions," in *Human-Robot Interaction (HRI), 2011 6th ACM/IEEE International Conference on*. IEEE, 2011, pp. 433–440.

[18] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[19] Video, "Video accompanying this paper, available online," http://kormushev.com/goto/IROS-2013, 2013.