Unsupervised Linking of Visual Features to Textual Descriptions in Long Manipulation Activities

Eren Erdal Aksoy[1], Ekaterina Ovchinnikova[1], Adil Orhan[1], Yezhou Yang[2] and Tamim Asfour[1]

***Abstract*— We present a novel unsupervised framework, which links continuous visual features and symbolic textual descriptions of manipulation activity videos. First, we extract the semantic representation of visually observed manipulations by applying a bottom-up approach to the continuous image streams. We then employ a rule-based reasoning to link visual and linguistic inputs. The proposed framework allows robots (1) to autonomously parse, classify, and label sequentially and/or concurrently performed atomic manipulations (e.g. "*cutting*" or "*stirring*"), (2) to simultaneously categorize and identify manipulated objects without using any standard feature-based recognition techniques, and (3) to generate textual descriptions for long activities, e.g. "*breakfast preparation*". We evaluated the framework using a dataset of 120 atomic manipulations and 20 long activities.**

I. Introduction

Integration of textual descriptions and visual features has gained an increasing attention in natural language processing, computer vision, and robotics ([1]–[6]). The main challenge consists in bridging the gap between perceived continuous visual features and discrete symbolic linguistic constructions. In the context of robot learning from demonstration, this problem is called *symbol grounding* [7] referring to the grounding of the observed high-level symbolic action or object concepts into the low-level sensory-motor data.

In this work, we introduce a novel unsupervised method allowing the discretization of visually observed continuous manipulations into the high-level symbolic object-action concepts which can be directly mapped to their counterparts in linguistic video descriptions. This helps robots to ground language in vision, i.e. makes the *symbol grounding* problem treatable. For instance, in human-robot interaction tasks, e.g. *dinner preparation*, robots can autonomously parse individual atomic actions and the role of each manipulated tool and associate them with human instructions. Thus, robots can learn how each visual entity in the continuous demonstration is described in language, e.g. *"a big knife"*, or what a symbolic action command *"cut"* means in terms of interactions between visual entities. Likewise, given a set of human demonstrations, robots can further learn the most probable atomic action sequence, e.g. a long-term action plan to prepare the dinner. As robots perform the planned actions, they can also verbally inform the user about each action.

State of the art methods approach this problem by applying pre-trained object and action detectors to the observed demonstrations. We, however, do not employ object or action recognition, since it is not feasible to introduce every possible tool or action type that robots may encounter. Instead, our framework allows robots to autonomously categorize a) demonstrated actions based on their semantic similarities and b) manipulated objects based on their roles in actions. Thus, our framework neither requires prior object or action knowledge, nor performs action or object recognition in the traditional sense. The framework assigns linguistic labels to the acquired action and object concepts by extracting them from textual descriptions of the demonstrations.

Our framework has two phases: learning and testing. In the learning phase, we employ the "Semantic Event Chain" (SEC) concept [8], which captures the semantic representation of a continuous manipulation by discretizing it into states (Sec. III-A.2). A pattern of SEC states forms symbolic action concepts and suggests object categories based on their roles in the manipulation. Learning is concluded by labeling these symbolic object and action concepts with linguistic labels extracted from textual video descriptions. In the testing phase, we analyze chained manipulation sequences, which are decomposed into sequentially and/or concurrently performed manipulation instances. We compute object and action concepts for each instance and match them with the learned labeled concepts to generate textual descriptions.

We applied our framework to a large publicly available manipulation action (ManiAc) dataset [9] with 120 demonstrations of 8 manipulation types (e.g. "*cutting*") and 20 long activities (e.g. "*breakfast preparation*"). We used the Mechanical Turk (MTurk) crowdsourcing platform to acquire 5 textual descriptions for each video. In the experiments on the ManiAc dataset, our approach outperformed the standard action recognition methods. We also obtained promising results on the description generation for long activities.

Contributions: (1) We introduce a novel unsupervised method for grounding language in vision by linking low-level visual sensory data to their symbolic descriptions. (2) We label visual object and action concepts by using their textual descriptions only, without employing any standard feature-based recognition techniques. (3) We generate a textual description for an unseen complex activity, in which various manipulations are chained sequentially or concurrently. (4) We evaluate our framework on a large dataset.

II. State of the Art

Related literature on linking natural language and vision mostly focuses on either generating textual descriptions for

This work has been supported by the EU FET Proactive grant (GA: 641100) TIMESTORM.

[1]H[2]T, Karlsruhe Institute of Technology, Germany.
[2]CIDSE, Arizona State University, AZ, USA.

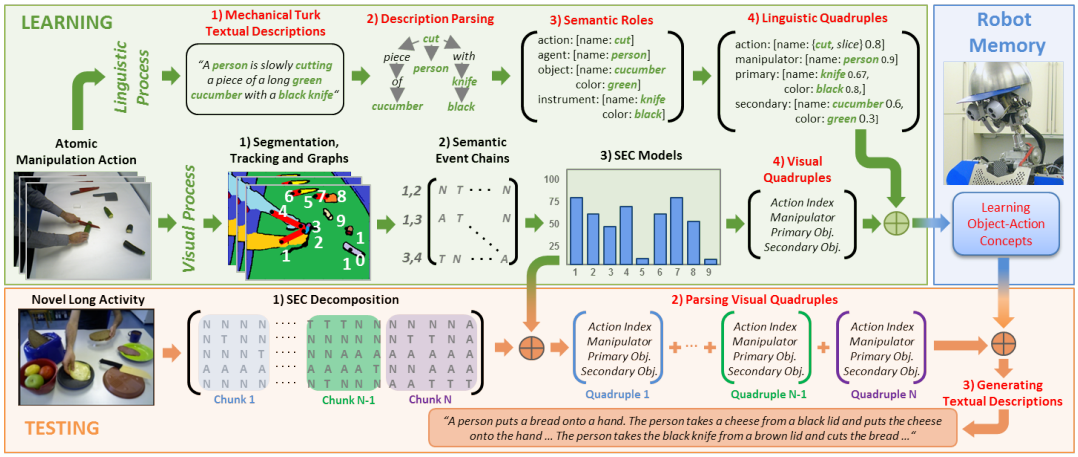

Fig. 1. Overview of the proposed framework. Headers written in red indicate the main contribution of this work.

static scenes (e.g. [10], [11]) or learning linguistic descriptions only for object concepts (e.g. [4]), without considering action information. In [1]–[3], [5], [12], textual descriptions of videos are generated by employing supervised visual detectors. Semantic hierarchies are used to generate the descriptions by detecting pre-trained $\langle subject, object, action \rangle$ triplets [1]. A similar data-driven approach in [2] considers the best two visual object detector results as *subject* and *object* candidates. In [12], a more accurate probabilistic factor-graph model is created by including the scene (location) information. The recent method along these lines uses Hidden Markov Models (HMMs) to learn a statistical relation between followed motions, manipulated objects, and their manually assigned linguistic descriptions [5]. In [13], HMMs are also used for recognizing events based on motion profiles. In [14], discriminative models (e.g. Conditional Random Fields) predict semantic representations of the visual content, which are then translated into textual descriptions.

All these approaches analyze short videos and rely on visual feature-based object and action descriptors (e.g. SIFT, STIP, improved dense trajectories), which require large sets of labeled training data. Such methods are limited to known object-action classes and suffer from the generalization problem if training data are limited. Unlike these methods, our unsupervised SEC approach considers only the roles of the manipulated objects to learn object and action categories, without employing any conventional visual object or action detector. In our framework, structured linguistic information is further employed to label these learned categories, whereas the aforementioned approaches use linguistic information to tackle the problem of inaccurate or missing visual detections. In this respect, our approach bears similarities to [15] which introduces a hyperfeature coding of videos to be aligned with linguistic data using unsupervised hand and object detection.

In the context of robotics, HMMs are used to represent human whole-body motion data by considering the joint angle or position data only, without incorporating object information [6]. Although this method supports a bi-directional mapping between human whole-body motion and linguistic utterances, using only motion data limits the application of the method to manipulations, in which the motion profiles of objects and hands vastly vary between demonstrations. In addition, classical HMM-based approaches are not suitable for recognizing parallel streams of actions [16] and cannot easily describe motions with repetitions or recursions [17]. SECs obey the Markovian assumption, but in contrast to HMM-based generative frameworks, the states in SECs are observable and represent topological changes in the scene.

Another line of work uses deep networks for generating video descriptions. In [3], recurrent neural networks are used to map image sequences to word sequences, after being trained on raw images and optical flow data. In [18], video descriptions are generated by fusing appearance and optical flow cues extracted from a spatio-temporal convolutional network. Although these approaches show promising results on unconstrained videos, they omit the action semantics and require tuning of a large set of hyper-parameters. Since they depend on the the motion features, they require a large training set, because optical flow features can vary significantly even between the same action demonstrations. In contrast, our approach extracts only descriptive spatio-temporal discontinuities in relations between objects and subjects in the scene, which remain stable for a given manipulation type, even with large motion and object variations.

III. Our Approach

We refer to our system as "ViLaRob" standing for "coupling Vision and Language for Robotics". The proposed framework shown in Fig. 1 consists of two processing phases: learning and testing, results of which are combined in a robot memory. In the learning phase (Fig. 1 green box), we process visual and linguistic features extracted from an input atomic manipulation video annotated with textual descriptions. In the visual process, the scene content is encoded by graphs derived from tracked unique image segments. Graphs are converted into SECs which are used to learn action concepts, i.e. SEC models, in an unsupervised way. Given the SEC, the framework estimates an action concept index by

comparing it with the learned SEC models and categorizes the tracked image segments as *manipulator*, *primary* and *secondary objects*. These action and object information are stored as a *visual quadruple*. In the linguistic process, we use the MTurk platform to obtain textual descriptions for each video. We parse the textual descriptions and extract labels of the action and its participants (*semantic role* fillers). The extracted label set for each video is converted into a *linguistic quadruple*. Learning is concluded by combining visual and linguistic quadruples. The combined data is stored as Object-Action Concepts (OACs) in the robot memory (Fig. 1 blue box). OACs are co-joint symbolic representations of the robot's sensorimotor experience, which is very much related to the object-action complexes introduced in [19].

In the testing phase (Fig. 1 orange box), we process videos showing novel long manipulation activities, e.g. "*making a sandwich*". The process starts with extracting the long SEC representation, which is decomposed into chunks by considering the interaction between the estimated *manipulator* and objects in the scene. Each SEC chunk is compared to the learned SEC models to estimate an action concept index. After applying the object role categorization, we generate visual quadruples for each chunk. The framework then compares these quadruples with the learned OACs in the robot memory, assigns labels to objects and actions, and generates a linguistic description for the novel long activity.

In the following, we detail processing steps in Fig. 1, where black headers represent modules inherited from our previous works [8], [9], [20], [21], whereas red headers indicate our main contributions in the ViLaRob framework.

A. Learning Stage: Visual Process

1) Image Segmentation, Tracking, and Graphs: The input of our framework is an *RGB-D* image stream of a human manipulation demonstration. After applying a color and depth-based image segmentation [22], we track all objects and hands in the scene. Each segmented image is converted into a graph, in which nodes show segment centers and edges represent the contact (i.e. touching) relation between segment pairs. In Fig. 2, some images with the tracked image segments and graphs are shown for a sample *cutting* action.

2) Semantic Event Chains (SECs): Given a continuous graph sequence, an exact graph matching method is applied to extract a set of main graphs representing topological changes in the scene. Main graphs are used to construct the SEC matrix (ψ). SEC rows describe spatial relations between two objects and columns encode the topological scene structure at each main graph. Spatial relations in the SEC rows are *Not touching (N)*, *Touching (T)*, and *Absence (A)*. Fig. 2 shows the SEC matrix for a *cutting* example. For instance, the third SEC row represents the spatial relations between nodes 9 and 7, i.e. the left hand and the knife. The third SEC column indicates the state when the left hand starts grasping the knife. The SEC matrix encodes object pairs that produce at least one relational change (e.g. from N to T) and ignores those with static relations (e.g. between the left and right hands). The SEC concept was introduced in [8].

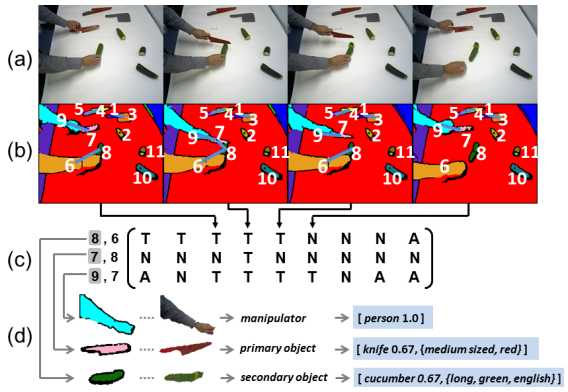

Fig. 2. A sample *cutting* action with (a) original images, (b) segments and main graphs. Numbers represent graph nodes and edges are given by blue lines. (c) The SEC matrix in which N represents two disjoint object segments, T shows a touching relation between two objects, and A is for the absence of an object. (d) Extracted *manipulator*, *primary* and *secondary objects* with corresponding linguistic labels (shown in blue boxes).

3) Learning SEC Models: Given a set of human demonstrations, we employ the unsupervised learning framework introduced in [9] to learn an action concept, i.e. a SEC model matrix (ψ^{m}). The learning method works in an online fashion and is initiated once a new manipulation is observed. For instance, when the first atomic manipulation is demonstrated, the extracted SEC sample ψ_1 is treated as the first model ψ_1^{m} and stored in a library. Once the i-th demonstration is shown, we encode it again by a SEC ψ_i and measure the semantic similarities $\zeta(\cdot,\cdot)$ with all existing SEC models in the library. The similarity between a SEC sample and a model, i.e. $\zeta(\psi_i, \psi_j^{\mathrm{m}})$, is computed by comparing rows and columns as described in [8]. If the computed maximum similarity is higher than an automatically estimated threshold [9], the new SEC sample ψ_i is assigned to the most similar model $\psi_\Lambda^{\mathrm{m}}$, where $\Lambda = \arg\max_{1\leq j\leq \eta}(\zeta(\psi_i, \psi_j^{\mathrm{m}}))$ and η is the total model number in the library. The model $\psi_\Lambda^{\mathrm{m}}$ is then updated with additional rows or columns that might exist in ψ_i. In this way, the SEC models will only consist of those rows and columns that are frequently observed in all assigned SEC samples. If the maximum similarity is lower than the threshold, the SEC sample ψ_i is introduced as a new action model, i.e. $\psi_{\eta+1}^{\mathrm{m}}$. All learned SEC models are stored in the library as a set of action concepts $\Psi^{\mathrm{m}} = \{\psi_1^{\mathrm{m}}, \cdots, \psi_{\eta+1}^{\mathrm{m}}\}$.

4) Visual Quadruples: We assign an action concept index to a SEC sample and categorize SEC graph nodes, i.e. image segments, according to their roles in the action. The action concept index, Λ, of the SEC sample is the index number of the best fitting SEC model computed in section III-A.3. For instance, the SEC sample in Fig. 2 has the highest similarity with the first model ψ_1^{m}, thus, $\Lambda = 1$.

To categorize SEC graph nodes, we employ the method in [20], which assumes that each manipulation involves three main participants: *manipulator* (M), *primary* (P) and *secondary* (S) objects. The *manipulator*, M, e.g. a *hand*, frequently interacts with other objects in the scene and is estimated as the graph node that participates in most of the relational changes (e.g. from N to T) and has the longest

touching relations in the SEC. The object, which has the longest touching relations with M, is then considered as the *primary object* P, e.g. a knife in the *cutting* action. *Secondary objects*, S, represent those nodes that interact, i.e. have touching relations, with P, e.g. a cucumber to be cut. In Fig. 2, nodes 9, 7, and 8 are respectively categorized as M, P, and S. This categorization method does not rely on object recognition, but applies a rule-based reasoning to identify the naked graph nodes based on changes in their spatio-temporal contact relations (see [20]). Although here we focus on uni-manual manipulations only, the framework can be extended to bi-manual manipulations.

The estimated SEC action concept index and object categories from each manipulation video are stored as a *visual quadruple*, i.e. $Q_V = (\Lambda, M, P, S)$, to be later matched with linguistic counterparts. In Fig. 2, the obtained visual quadruple is $Q_V = (\Lambda = 1, M = 9, P = 7, S = 8)$.

B. Learning Stage: Linguistic Process

We now describe the fully automatic unsupervised generation of linguistic quadruples. We assume that the conceptualization of the visual input expressed in natural language corresponds to our visual encoding of the action by visual quadruples, i.e. elements of visual quadruples are described by certain syntactic structures in language.

1) Mechanical Turk Textual Descriptions: For action videos, we obtain textual descriptions (e.g. Fig. 5), which are processed to generate symbolic labels for the action, manipulator, and manipulated objects. The descriptions can be grammatically incorrect and contain typos. Therefore, we do not rely on one description per video, but obtain several of them and extract most frequent labels as described below.

2) Description Parsing: First, each textual description is parsed in order to abstract from syntactic variations. In the experiment described below, we use the *Boxer* parser [23].

3) Semantic Roles: In language, verbs often refer to actions. We extract verbs from the parsed description as potential labels for the action shown in the video. Each action has participants, e.g. an agent, a manipulated object, an instrument. In language, the participants are often described by noun phrases linked to the verb syntactically. Each participant plays a certain role in the action called a *semantic role* [24]. For each verb, we extract fillers (noun phrases) of the semantic roles of

- *agent* (deliberately performing the action, e.g. *person*),
- *patient* (undergoing the action, e.g. *carrot* in cutting),
- *instrument* (the main tool, e.g. *knife* in cutting), and
- *location* (where the action occurs, e.g. *bowl* in stirring).

We assume that agents are expressed by syntactic subjects, patients – by direct objects, instruments – by prepositional phrases with the instrument prepositions (e.g. *cutting* ***with*** *a knife*) or by the verb *use* and its synonyms (***using*** *a knife* ***for*** *cutting*), and locations – by prepositional phrases with location prepositions (e.g. *on*, *from*, *at*). We also handle constructions like `PART of NOUN` (e.g. *piece of cucumber*), where `PART` is any noun referring to a part of an object (e.g. *piece*, *slice*), and `NOUN` as the corresponding object label (e.g. *cucumber*). Based on these assumptions, we formulate rules for automatically extracting nouns as potential labels for objects participating in the action. For finding synonyms and location/instrument prepositions, we use lexical databases WordNet and FrameNet.[1] The mentioned patterns cover most frequent syntactic constructions, but are not exhaustive. Corpora mining can be employed to further expand patterns.

A role filler may be expressed by a pronoun or not linked to the action verb syntactically, e.g. *knife* is not linked to *cut* in *"A person grasps a knife and cuts a cucumber"*. To resolve the reference or detect missing role fillers which are not covered by the above patterns, we select the noun in the sentence that is most likely to be a filler of the corresponding role. For this purpose, we use a database of dependency tuples extracted from a large corpus [25]. A dependency tuple consists of a syntactic relation type, the fillers of the relation, and the frequency of their co-occurrence in a corpus, e.g. ⟨*prep phrase*, *cut*, *with*, *knife*, 7399⟩. For each action verb without a role filler, we select nouns with the highest frequencies to be the role fillers in the database.

We also extract action and object descriptors (*slow*, *gently*, *red*, *long* etc.) expressed by adverbs and adjectives. As a result, for each textual description, we obtain several semantic structures, each containing a name and descriptors for *action*, *agent*, *patient*, *instrument*, and *location*.

4) Linguistic Quadruples: For each video, the semantic structures are converted into quadruples in three steps. First, we calculate frequencies of each linguistic action label (Λ') in the semantic structures, so that synonyms in the WordNet database are considered to be the same label, e.g. *cut* and *slice*. Second, we merge semantic structures that share the action label and calculate frequencies of each role filler and their descriptors. Third, we map the fillers of the roles *agent*, *patient*, *instrument*, and *location* to *manipulator* (M'), *primary* (P') and *secondary* (S') objects. Agents are directly mapped to M'. If an instrument filler has a higher frequency than a location filler, the instrument filler is mapped to P' and the patient filler is mapped to S', e.g. in *"cut a cucumber with a knife"*, *knife* and *cucumber* are mapped to P' and S', respectively. Otherwise, the patient filler is mapped to P' and the location is mapped to S', e.g. in *"put a box on a cup"*, *box* and *cup* are linked to P' and S', respectively.

We finally store the action label and the role fillers for each video description as a *linguistic quadruple*, i.e. $Q_L = (\Lambda', M', P', S')$. For the example in Fig. 2 the obtained quadruple is $Q_L = (\Lambda' = \text{“}cut\text{”}, M' = \text{“}person\text{”}, P' = \text{“}knife\text{”}, S' = \text{“}cucumber\text{”})$. Note that each element of Q_L is assigned a normalized frequency value and attributes.

C. Robot Memory: Object-Action Concepts (OACs)

We now match the elements of quadruples Q_L and Q_V in order to find the most probable linguistic labels for the learned SEC models and graph nodes, i.e. image segments in SECs. The most frequent action label Λ' in Q_L is directly

[1] *https://wordnet.princeton.edu*, *https://framenet.icsi.berkeley.edu*

mapped to the SEC action index Λ in Q_V, e.g. in Fig. 2, the label "*cut*" is mapped to the first SEC model $\psi_1^{\mathfrak{m}}$. Likewise, M', P', and S' in Q_L are respectively mapped to their visual counterparts in Q_V, cf. Fig. 2. Thus, without applying object recognition, we can label, for instance, the pink segment, i.e. node 7, in Fig. 2 as "*knife*" with the attributes "*medium sized*" and "*red*". The final matched data between Q_L and Q_V form a set of Object-Action Concepts (OACs) as $\Omega_{OAC} = \{\cdots, (\Lambda_i, \Lambda_i'), (M_i, M_i'), (P_i, P_i'), (S_i, S_i'), \cdots\}$ where i is the observed manipulation number in the learning stage. The learned Ω_{OAC} is stored in the robot memory to identify new visual quadruples detected in the testing stage.

D. Testing Stage

1) SEC Decomposition: The testing phase starts with an analysis of an unseen long manipulation video, e.g. showing "*making a sandwich*", which needs to be first temporally decomposed into chunks to detect each sequentially or concurrently performed atomic action such as *cutting* or *stirring*.

First, we extract the event chain of a long manipulation sequence. The decomposition process is triggered with the estimation of the *manipulator* M (see section III-A.4) from the long SEC. Next, we apply the method introduced in [20], which searches for $[N, T]$ and $[T, N]$ relational changes in SEC rows that involve the *manipulator* M. These changes are cutting points; a change from N to T indicates the start point, whereas a change from T to N defines the end point of the manipulation. For instance, when a hand grasps a knife, the relation in the corresponding SEC row switches from N to T. These semantic relational changes indicate potential temporal borders, in which atomic actions take place.

Fig. 3 (a) depicts the event chain for a sample manipulation sequence, in which a hand is removing a cup, putting an apple down, and hiding it with the cup. In this example, the segment 7 is correctly estimated as the *manipulator*. Colored blocks in the SEC highlight sequences of $[N, T, \cdots, T, N]$

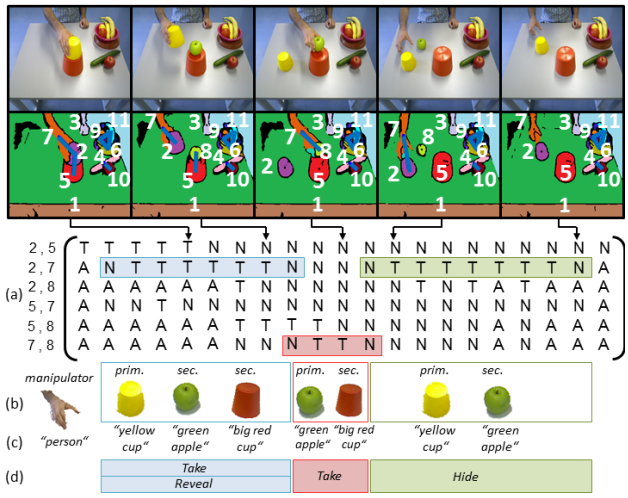

Fig. 3. (a) SEC with $[N, T, \cdots, T, N]$ colored blocks, i.e. unique SEC chunks. (b) Visual *manipulator*, *primary* and *secondary object* segments in SEC chunks. (c) Linguistic object labels. (d) Mapped manipulation labels.

relations that belong to the *manipulator*. These blocks represent the start and end points of the three unique SEC chunks.

2) Parsing Visual Quadruples: Once the long SEC Ψ is decomposed, we obtain sequentially performed k different SEC chunks as $\Psi = \{\psi_1, \cdots, \psi_k\}$. In each chunk ψ_i, we employ the technique described in Sec. III-A.4 and search for the *primary* (P_i) and *secondary* (S_i) objects. We assume that each chunk ψ_i involves at most one *primary object*, since a hand can grasp and manipulate only one object at a time. However, there might be n different *secondary objects*, i.e. $S_i = \{s_i^j : j \in [1, \cdots, n]\}$. We treat each s_i^j as an indicator of a potential parallel action stream because, if different manipulations share the same temporal interval, each has to have a unique *secondary object*. We apply the brute force combinatorial process in [20], which assesses each set of $\{M, P_i, s_i^j\}$ as one manipulation hypothesis ψ_i^j. Each hypothesis is compared with the learned SEC models in $\Psi^{\mathfrak{m}}$ to explore the best matching action concept index Λ_i^j.

Finally, we construct visual quadruples for each hypothesis as $Q_V^{\psi_i^j} = (\Lambda_i^j, M, P_i, s_i^j)$. To identify object and action labels, we compare each $Q_V^{\psi_i^j}$ with the already labeled visual quadruples, i.e. the learned Ω_{OAC} (see Sec. III-C). The comparison of objects is based on matching the color and texture features of the image segments in $Q_V^{\psi_i^j}$ with that of stored in Ω_{OAC}. The labels of the best matched elements in Ω_{OAC} are assigned to the corresponding elements in $Q_V^{\psi_i^j}$. We emphasize that this feature comparison step is not employed for any object detection purpose, but to search for the most similar segments in the memory. This step is not a contribution of the framework; any other matching method can be employed instead.

Fig. 3 (b) shows the estimated P_i and S_i in each SEC chunk. For instance, in the temporal interval of the blue block in Fig. 3 (a), segment 2 is estimated as P_1, since it has the most touching relations with the previously detected M (segment 7). Segments 5 and 8 are estimated as *secondary objects* s_1^1 and s_1^2, since they are the only segments touching to P_1 in the blue block. Thus, the framework returns two parallel manipulation hypotheses: $\psi_1^1 = \{7, 2, 5\}$ and $\psi_1^2 = \{7, 2, 8\}$, which are most similar to models $\psi_{\Lambda_1}^{\mathfrak{m}}$ and $\psi_{\Lambda_2}^{\mathfrak{m}}$. Fig. 3 (c) shows the matched linguistic labels obtained by comparing these object segments with those in Ω_{OAC}. M is labeled as "*person*" and P_1 as "*yellow cup*". *Secondary objects* s_1^1 and s_1^2 are labeled as "*green apple*" and "*big red cup*". Fig. 3 (d) shows the best matched action labels (Λ_i') for each chunk, e.g. the framework detects two parallel actions labeled as "*take*" and "*reveal*" in the blue SEC chunk.

3) Generating Textual Descriptions: The labeled visual quadruples are used for generating a textual description. For each visual quadruple, we generate a sentence "*Determiner (A, The) + manipulator + action verb (3rd person present) + primary object description + preposition + secondary object description*". *Object description* is given as "*determiner (a, the) attributes (adjectives) object noun*". The positions of *primary* and *secondary object descriptions* can be interchangeable depending on whether a location or an instrument

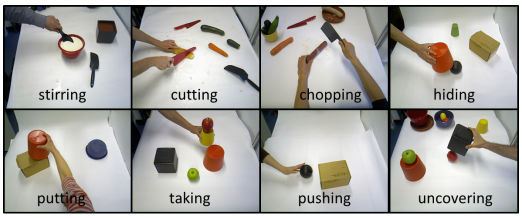

Fig. 4. Sample frames for eight different actions in the ManiAc dataset.

preposition is more frequently used with the action verb in the MTurk descriptions. For example, the following sentence is generated for the action sequence in Fig. 3: "*A person takes a yellow cup from a big red cup and reveals a green apple hidden under the yellow cup. The person takes the green apple from the big red cup. The person hides the green apple with the yellow cup.*" Unknown actions and objects are labeled with "*manipulate*" and "*something*", respectively.

IV. Experimental Evaluation

For the evaluation, we used the publicly available manipulation action dataset ManiAc[2] [9] with eight atomic manipulation types: *stirring*, *cutting*, *chopping*, *hiding*, *putting*, *taking*, *pushing*, and *uncovering*. In contrast to other datasets, MainAc focuses on human-object interactions recorded as RGB-D image streams and has high intra-class variations. For each type, there are 15 videos, with five human subjects manipulating 30 objects. Fig. 4 shows a sample frame for each manipulation type. We used these, in total, 120 manipulations for the learning phase of our framework.

A. Learning Phase

To obtain textual descriptions for atomic action videos, we employed the Amazon Mechanical Turk platform. In each MTurk task description, we showed a video and the instruction: *Watch the video and describe actions and manipulated objects in the video and their aspects. Example description: A person is gently cutting a green long cucumber with a red knife*. We collected five descriptions per video, i.e. 600 descriptions for 120 videos in total. We obtained a significant variance between acquired textual descriptions. Fig. 5 shows example descriptions and the extracted linguistic quadruple Q_L for a cutting video. We obtained one Q_L per video.

Given the 120 atomic videos, we extracted the corresponding SEC representations and learned SEC models using

[2] *https://fortknox.physik3.gwdg.de/cns/index.php?page=maniac-dataset*

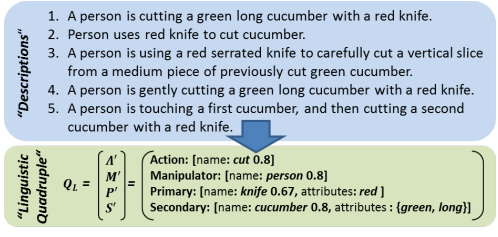

Fig. 5. Collected descriptions and extracted linguistic quadruple.

TABLE I

Percentage of described action labels for each SEC model.

Λ	ψ_1^m	ψ_2^m	ψ_3^m	ψ_4^m	ψ_5^m	ψ_6^m	ψ_7^m
Λ'	**cut: 88** chop: 4 slice: 4 saw: 4	**stir: 100**	**move: 79** slide: 7 put: 7 push: 7	**hide: 76** close: 8 cover: 8 put: 8	**take: 76** move: 8 put: 8 remove: 8	**put: 92** place: 8	**reveal: 68** uncover: 8 take: 8 remove: 8 lift: 8

the online learning approach (Sec. III-A.3). Our learning method retrieved 7 SEC models, i.e. action concepts, for 8 manipulation types, such that $\Psi^{\mathfrak{m}} = \{\psi_1^{\mathfrak{m}}, \cdots, \psi_7^{\mathfrak{m}}\}$. This is because the learning approach merged *cutting* and *chopping* samples and generated one single model for both types. Those two manipulations are indeed similar since both yield the same consequence, i.e. *splitting* objects into parts. Primitives, i.e. SEC columns, in both action types are the same. Differences are the followed motion and velocity profiles of the movements, which are not captured by SECs.

All 120 SEC samples were compared with these 7 models in $\Psi^{\mathfrak{m}}$ to predict the action concept index Λ. After categorizing the manipulated objects as M, P, and S, we extracted the Q_V representation of each video and matched them with the Q_L counterparts to create Ω_{OAC}. Table I shows the distribution of the matched linguistic labels (Λ') for each SEC model in $\Psi^{\mathfrak{m}}$. For example, model $\psi_1^{\mathfrak{m}}$ was learned from both *cutting* and *chopping* video samples which were mostly (88%) described as "*cut*", but rarely as "*chop*","*slice*", or "*saw*" by the MTurk annotators. Having mostly the same symbolic labels for the *cutting* and *chopping* videos supports our claim that these actions are semantically similar.

Fig. 6 shows the labeled *manipulator*, *primary* and *secondary objects* for actions: "*cut*" and "*stir*". For instance, our approach learned that the *primary object* for *cutting* is mostly labeled as "*knife*" with various attributes, e.g. "*medium sized red*" or "*black*". The same label was also learned for the *primary object* in *stirring*. Indeed, in the ManiAc dataset, some subjects selected knives as the tool for *stirring*. Note that such a top-down reasoning plays a vital role for cognitive robots to explore grounded object affordances.

We also manually validated the generation of linguistic and visual quadruples in the learning phase, see Fig. 7. For

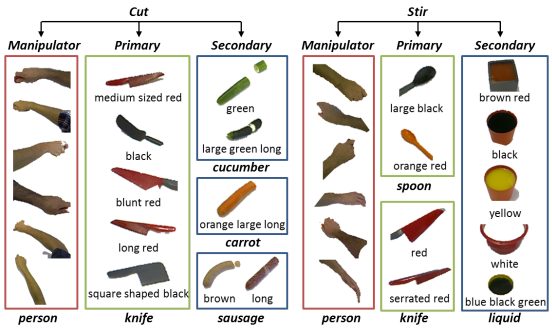

Fig. 6. Learned object labels for actions *cut* (left) and *stir* (right). Each box indicates a different concept, labels of which are given beneath. Attributes of each object are given under the corresponding image, if there are any.

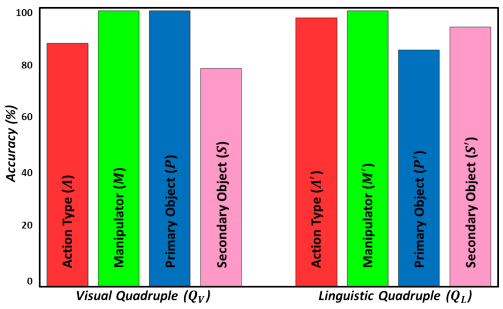

Fig. 7. Accuracy measures in the visual and linguistic quadruple generation in the learning phase. Note that for linguistic quadruples, we validated verb and noun labels only; attributes (adjectives and adverbs) were ignored.

the visual quadruples, the accuracy indicates the percentage of correctly estimated (true-positive) action concept indices and object roles. For the linguistic quadruples, the accuracy is the percentage of linguistic labels that we validated as acceptable for describing objects and actions in the video.

B. Testing Phase

ManiAc also provides 20 long chained activities, e.g. "*making a sandwich*" or "*preparing a breakfast*". These activities are composed of over 100 different versions of the 8 atomic actions and some novel action types, e.g. *pouring*, which were not seen in the learning phase. Atomic actions are presented sequentially or concurrently in different scene contexts. We used these 20 long activities for testing.

First, the 20 activities were converted into SECs, each of which was decomposed into smaller chunks (Sec. III-D.1). In each SEC chunk, we parsed all possible action hypotheses by categorizing the objects as M, P, and S (Sec. III-D.2). An action concept index Λ was assigned to each hypothesis after comparing the hypotheses with the 7 models in Ψ^{m}. The predicted action indices and temporal borders of SEC chunks were then compared with the ground truth provided in the dataset to measure the action decomposition and classification accuracies, i.e. the average true positive rates. The average decomposition and classification accuracies are 90.9% and 84.9%, respectively. These results are comparable with our previous findings reported in [20], [21].

To generate textual descriptions for the 20 long activities, we computed the Q_V representation of each hypothesis and matched them with the already labeled data in Ω_{OAC}. This matching process returns linguistic labels for the action and objects in the hypotheses. Note that novel objects that were unseen in the atomic videos but appeared in the long videos were manually appended to the OACs with their linguistic labels. By employing the matched linguistic descriptors in all hypotheses (Sec. III-D.3), we generated in total 104 textual descriptions. Fig. 8 shows the qualitative results of automatically generated descriptions for three long activities.

We asked three validators to label each sentence, action verb, and object noun phrase as *correct* (all description parts are correct), *partially correct* (some parts are correct), and *wrong* (all parts are wrong), cf. Fig. 8. Table II shows

TABLE II

AVERAGE ACCURACY OF GENERATED DESCRIPTIONS.

	One		Two		All Three	
	C	C+P	C	C+P	C	C+P
Sentence description	.61	.95	.56	.93	.42	.91
Action verb	.92	.94	.91	.93	.75	.91
Primary object noun	.70	.82	.64	.77	.61	.73
Secondary object noun	.74	.89	.70	.84	.68	.80

percentages of the descriptions that are correct (C) and correct or partially correct (C+P) according to any one, any two, or all three validators. The inter-annotator agreement Fleiss' kappa is 0.76. The pairwaise unweighted Cohen's kappa is 0.73, 0.83, and 0.71 for each pair of validators. These values show that the three validators have a high level of agreement. We also asked the validators to indicate the number of atomic actions that were not described, which is measured as 0.26 per video.

Most errors in the generated descriptions resulted from the errors in the object matching. Lack of context sensitivity was also an issue. The same manipulation action might need to be labeled differently depending on the manipulated objects. For example, if a bowl is used to cover an apple, it can be described as *hiding an apple with a bowl*. But sandwich making cannot be described as *hiding the bread with cheese*. Another issue concerns composite entities. If a piece of cheese is put on the bread, the resulting object is a sandwich, while our method still calls it *cheese*.

The runtime of the SEC model learning is about 25 mins, although the segmentation, tracking, and SEC extraction run in real-time (25 Hz) on a PC with Intel Core i7 3.33 GHz CPU with 11.8 GB RAM and an Nvidia card GTX 295. Linguistic semantic roles are generated in around 6 ms per textual description on average, while linguistic quadruples are generated in about 45 ms per video on average. In the testing phase, the average temporal decomposition and recognition time is about 8 secs per video.

V. CONCLUSION

We introduced a novel framework for grounding language in vision by bridging the gap between continuous visual information and discrete linguistic descriptions of manipulation activities. The framework allows robots to identify and learn co-joint object-action concepts without prior knowledge.

The proposed framework identifies three action participants: *manipulator*, *primary* and *secondary objects*. This is clearly a limitation from a cognitive point of view, because humans distinguish between more action components, e.g. instrument, location, destination, which are reflected in the linguistic descriptions of the scenes. We plan to extend our visual and linguistic perception to cover more action components for a better cognitive approximation. Another limitation concerns the accurate segment tracking and object matching methods. Since both are not in the focus of this study, their naive implementations inject noise to the generated video descriptions. We plan to employ more advanced computer vision methods for the improvement. Context sensitivity and

Fig. 8. Three long activities together with the automatically generated textual descriptions. Black bars represent start and end points of each detected atomic action. Green, blue, and red boxes respectively highlight correct, partially correct, and wrong descriptions. Words in red indicate false labels.

composite entities pose challenges for the linguistic process (Sec. IV-B). We will address these issues by (a) using more training data to learn object-dependent action labels and (b) applying world knowledge resources to relabel composite entities. Future work also concerns the execution of learned action concepts with robots by employing generic execution skills, cf. [26]. We also plan to introduce an interactive dialog allowing robots to ask human demonstrators for help in identifying *unknown* object or action concepts not stored in the OACs. Such human-robot interaction would boost the performance of our framework in novel scenes, e.g. any mismatched concept in the new scene can be directly corrected by the human. To evaluate the scalability of our framework, we plan to benchmark it with more manipulation datasets (e.g. [15]) from various domains.

REFERENCES

[1] S. Guadarrama, N. Krishnamoorthy, G. Malkarnenkar, S. Venugopalan, R. Mooney, T. Darrell, and K. Saenko, "Youtube2text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition," in *Proc. of ICCV*, 2013, pp. 2712–2719.

[2] N. Krishnamoorthy, G. Malkarnenkar, R. J. Mooney, K. Saenko, and S. Guadarrama, "Generating natural-language video descriptions using text-mined knowledge," in *Proc. of AAAI*, 2013, pp. 541–547.

[3] S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko, "Sequence to sequence-video to text," in *Proc. of ICCV*, 2015, pp. 4534–4542.

[4] T. Nakamura, T. Nagai, K. Funakoshi, S. Nagasaka, T. Taniguchi, and N. Iwahashi, "Mutual learning of an object concept and language model based on mlda and npylm," in *IROS*, 2014, pp. 600–607.

[5] Y. Yamada, W. Takano, and Y. Nakamura, "Statistical behavioral understanding by motion, object, and language," in *Int. Federation for the Promotion of Mechanism and Machine Science*, 2015.

[6] W. Takano and Y. Nakamura, "Statistical mutual conversion between whole body motion primitives and linguistic sentences for human motions," *IJRR*, vol. 34, no. 10, pp. 1314–1328, 2015.

[7] S. Harnad, "The symbol grounding problem," *Physica D: Nonlinear Phenomena*, vol. 42, pp. 335–346, 1990.

[8] E. E. Aksoy, A. Abramov, J. Dörr, K. Ning, B. Dellen, and F. Wörgötter, "Learning the semantics of object-action relations by observation," *IJRR*, vol. 30, no. 10, pp. 1229–1249, 2011.

[9] E. E. Aksoy, M. Tamosiunaite, and F. Wörgötter, "Model-free incremental learning of the semantics of manipulation actions," *RAS*, vol. 71, pp. 118 – 133, 2015.

[10] B. Z. Yao, X. Yang, L. Lin, M. W. Lee, and S. C. Zhu, "I2t: Image parsing to text description." *Proc. of IEEE*, vol. 98, no. 8, 2010.

[11] G. Kulkarni, V. Premraj, S. Dhar, S. Li, Y. Choi, A. C. Berg, and T. L. Berg, "Baby talk: Understanding and generating simple image descriptions," in *Proc. of CVPR*, 2011, pp. 1601–1608.

[12] J. Thomason, S. Venugopalan, S. Guadarrama, K. Saenko, and R. Mooney, "Integrating language and vision to generate natural language descriptions of videos in the wild," in *Proc. of COLING*, 2014, pp. 1218–1227.

[13] H. Yu, S. Narayanaswamy, A. Barbu, and J. M. Siskind, "A compositional framework for grounding language inference, generation, and acquisition in video," *JAIR*, vol. 52, pp. 601–713, 2015.

[14] M. Rohrbach, W. Qiu, I. Titov, S. Thater, M. Pinkal, and B. Schiele, "Translating video content to natural language descriptions," in *Proc. of ICCV*, 2013, pp. 433–440.

[15] Y. C. Song, I. Naim, A. A. Mamun, K. Kulkarni, P. Singla, J. Luo, D. Gildea, and H. Kautz, "Unsupervised alignment of actions in video with text descriptions," in *Proc. of IJCAI*, 2016, pp. 2025–2031.

[16] J. Graf, S. Puls, and H. Wörn, "Recognition and understanding situations and activities with description logics for safe human-robot cooperation," in *Proc. of COGNITIVE*, 2010, pp. 90–96.

[17] K. Lee, Y. Su, T.-K. Kim, and Y. Demiris, "A syntactic approach to robot imitation learning using probabilistic activity grammars," *RAS*, no. 12, pp. 1323 – 1334, 2013.

[18] L. Yao, A. Torabi, K. Cho, N. Ballas, C. J. Pal, H. Larochelle, and A. C. Courville, "Video description generation incorporating spatio-temporal features and a soft-attention mechanism," *CoRR*, 2015.

[19] N. Krüger, C. W. Geib, J. H. Piater, R. P. A. Petrick, M. Steedman, F. Wörgötter, A. Ude, T. Asfour, D. Kraft, D. Omrcen, A. Agostini, and R. Dillmann, "Object-action complexes: Grounded abstractions of sensory-motor processes." *RAS*, vol. 59, no. 10, pp. 740–757, 2011.

[20] E. E. Aksoy, M. J. Aein, M. Tamosiunaite, and F. Wörgötter, "Semantic parsing of human manipulation activities using on-line learned models for robot imitation," in *Proc. of IROS*, 2015, pp. 2875–2882.

[21] E. E. Aksoy, A. Orhan, and F. Wörgötter, "Semantic decomposition and recognition of long and complex manipulation action sequences," *International Journal of Computer Vision*, pp. 1–32, 2016.

[22] A. Abramov, K. Pauwels, J. Papon, F. Wörgötter, and B. Dellen, "Depth-supported real-time video segmentation with the kinect," in *Proc. of WACV*, 2012, pp. 457–464.

[23] J. Bos, "Wide-Coverage Semantic Analysis with Boxer," in *Proc. of STEP*, ser. Research in Computational Semantics, 2008, pp. 277–286.

[24] C. J. Fillmore, "The case for case," in *Universals in Linguistic Theory*, E. Bach and R. T. Harms, Eds., New York, 1968.

[25] E. Ovchinnikova, V. Zaytsev, S. Wertheim, and R. Israel, "Generating conceptual metaphors from proposition stores," *arXiv preprint arXiv:1409.7619*, 2014.

[26] M. Wächter, S. Ottenhaus, M. Kröhnert, N. Vahrenkamp, and T. Asfour, "The ArmarX Statechart Concept: Graphical Programing of Robot Behavior," *Frontiers in Robotics and AI*, vol. 3, p. 33, 2016.