A Hierarchical Pose-Based Approach to Complex Action Understanding Using Dictionaries of Actionlets and Motion Poselets

Ivan Lillo
P. Universidad Catolica de Chile
Santiago, Chile
ialillo@uc.cl

Juan Carlos Niebles
Stanford University, USA
Universidad del Norte, Colombia
jniebles@cs.stanford.edu

Alvaro Soto
P. Universidad Catolica de Chile
Santiago, Chile
asoto@ing.uc.cl

Abstract

In this paper, we introduce a new hierarchical model for human action recognition using body joint locations. Our model can categorize complex actions in videos, and perform spatio-temporal annotations of the atomic actions that compose the complex action being performed. That is, for each atomic action, the model generates temporal action annotations by estimating its starting and ending times, as well as, spatial annotations by inferring the human body parts that are involved in executing the action. Our model includes three key novel properties: (i) it can be trained with no spatial supervision, as it can automatically discover active body parts from temporal action annotations only; (ii) it jointly learns flexible representations for motion poselets and actionlets that encode the visual variability of body parts and atomic actions; (iii) a mechanism to discard idle or non-informative body parts which increases its robustness to common pose estimation errors. We evaluate the performance of our method using multiple action recognition benchmarks. Our model consistently outperforms baselines and state-of-the-art action recognition methods.

1. Introduction

Human action recognition in video is a key technology for a wide variety of applications, such as smart surveillance, human-robot interaction, and video search. Consequently, it has received wide attention in the computer vision community with a strong focus on recognition of single actions in short video sequences [1, 21, 29, 36]. As this area evolves, there has been an increasing interest to develop more flexible models that can extract useful knowledge from longer video sequences, featuring multiple concurrent or sequential actions, which we refer to as *complex actions*. Furthermore, to facilitate tasks such as video tagging or retrieval, it is important to design models that can identify the spatial and temporal spans of each relevant action. As an example, Figure 1 illustrates a potential usage scenario, where an input video featuring a complex action is automatically annotated by identifying its underlying atomic actions and corresponding spatio-temporal spans.

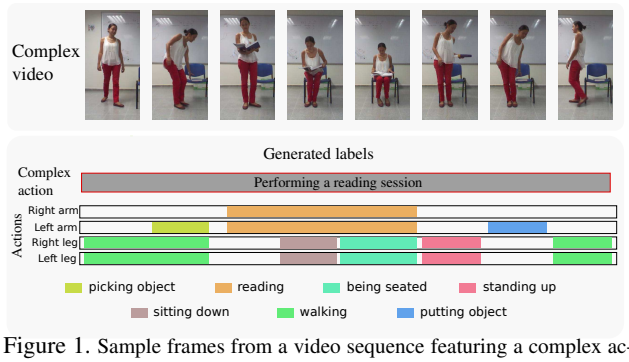

Figure 1. Sample frames from a video sequence featuring a complex action. Our method is able to identify the global complex action, as well as, the temporal and spatial span of meaningful actions (related to *actionlets*) and local body part configurations (related to *motion poselets*).

A promising research direction for reasoning about complex human actions is to explicitly incorporate body pose representations. In effect, as noticed long ago, body poses are highly informative to discriminate among human actions [13]. Similarly, recent works have also demonstrated the relevance of explicitly incorporating body pose information in action recognition models [11, 30]. While human body pose estimation from color images remains elusive, the emergence of accurate and cost-effective RGBD cameras has enabled the development of robust techniques to identify body joint locations and to infer body poses [25].

In this work, we present a new pose-based approach to recognizing and provide detailed information about complex human actions in RGBD videos. Specifically, given a video featuring a complex action, our model can identify the complex action occurring in the video, as well as, the set of atomic actions that compose this complex action. Furthermore, for each atomic action, the model is also able to generate temporal annotations by estimating its starting

and ending times, and spatial annotations by inferring the body parts that are involved in the action execution.

To achieve this, we propose a hierarchical compositional model that operates at three levels of abstraction: body poses, atomic actions, and complex actions. At the level of body poses, our model learns a dictionary that captures relevant spatio-temporal configurations of body parts. We refer to the components of this dictionary as *motion poselets* [2, 26]. At the level of atomic actions, our model learns a dictionary that captures the main modes of variation in the execution of each action. We refer to the components of this dictionary as *actionlets* [32]. Atoms in both dictionaries are given by linear classifiers that are jointly learned by minimizing an energy function that constraints compositions among *motion poselets* and *actionlets*, as well as, their spatial and temporal relations. While our approach can be extended to more general cases, here we focus on modeling atomic actions that can be characterized by the body motions of a single actor, such as running, drinking, or eating.

Our model introduces several contributions with respect to prior work [18, 26, 32, 34]. First, it presents a novel formulation based on a structural latent SVM model [39] and an initialization scheme based on self-pace learning [15]. These provide an efficient and robust mechanism to infer, at test and training time, action labels for each detected motion poselet, as well as, their temporal and spatial span. Second, it presents a multi-modal approach that trains a group of actionlets for each atomic action. This provides a robust method to capture relevant intra-class variations in action execution. Third, it incorporates a *garbage collector* mechanism that identifies and discards idle or non-informative spatial areas of the input videos. This provides an effective method to process long video sequences. Finally, we provide empirical evidence indicating that the integration of the previous contributions in a single hierarchical model, generates a highly informative and accurate solution that outperforms state-of-the-art approaches.

2. Related Work

There is a large body of work on human activity recognition in the computer vision literature [1, 21, 29, 36]. We focus on recognizing human actions and activities from videos using pose-based representations and review in the following some of the most relevant previous work.

The idea of using human body poses and configurations as an important cue for recognizing human actions has been explored recurrently, as poses provide strong cues on the actions being performed. Initially, most research focused on pose-based action recognition in color videos [8, 27]. But due to the development of pose estimation methods on depth images[25], there has been recent interest in pose-based action recognition from RGBD videos [7, 9, 28]. Some methods have tackled the problem of jointly recognizing actions and poses in videos [20] and still images [37], with the hope to create positive feedback by solving both tasks simultaneously.

One of the most influential pose-based representations in the literature is Poselets, introduced by Bourdev and Malik [3]. Their representation relies on the construction of a large set of frequently occurring poses, which is used to represent the pose space in a quantized, compact and discriminative manner. Their approach has been applied to action recognition in still images [19], as well as in videos [26, 33, 41].

Researchers have also explored the idea of fusing pose-based cues with other types of visual descriptors. For example, Cheron *et al*. [5] introduce P-CNN as a framework for incorporating pose-centered CNN features extracted from optical flow and color. In the case of RGBD videos, researchers have proposed the fusion of depth and color features [9, 14]. In general, the use of multiple types of features helps to disambiguate some of the most similar actions.

Also relevant to our framework are hierarchical models for action recognition. In particular, the use of latent variables as an intermediary representation in the internal layers of the model can be a powerful tool to build discriminative models and meaningful representations [10, 34]. An alternative is to learn hierarchical models based on recurrent neural networks [6], but they tend to lack interpretability in their internal layers and require very large amounts of training data to achieve good generalization.

While most of the previous work have focused on recognizing single and isolated simple actions, in this paper we are interested in the recognition of complex, composable and concurrent actions and activities. In this setting, a person may be executing multiple actions simultaneously, or in sequence, instead of performing each action in isolation. An example of these is the earlier work of Ramanan and Forsyth [22], with more recent approaches by Yeung *et al*. [38] and Wei *et al*. [35]. Another recent trend aims at fine-grained detection of actions performed in sequence such as those in a cooking scenario [24, 16].

We build our model upon several of these ideas in the literature. Our method extends the state-of-the-art by introducing a model that can perform detailed annotation of videos during testing time but only requires weak supervision at training time. While learning can be done with reduced labels, the hierarchical structure of poselets and actionlets combined with other key mechanisms enable our model to achieve improved performance over competing methods in several evaluation benchmarks.

3. Model Description

In this section, we introduce our model for pose-based recognition of complex human actions. Our goal is to build a model with the capability of annotating input videos with the actions being performed, automatically identifying the

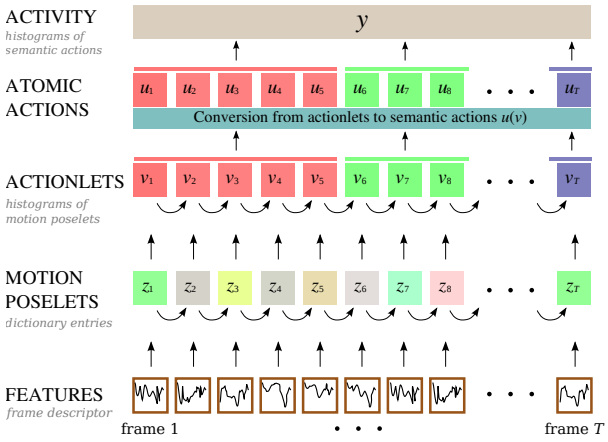

Figure 2. Graphical representation of our discriminative hierarchical model for recognition of complex human actions. At the top level, activities are represented as compositions of atomic actions that are inferred at the intermediate level. These actions are, in turn, compositions of poses at the lower level, where pose dictionaries are learned from data. Our model also learns temporal transitions between consecutive poses and actions.

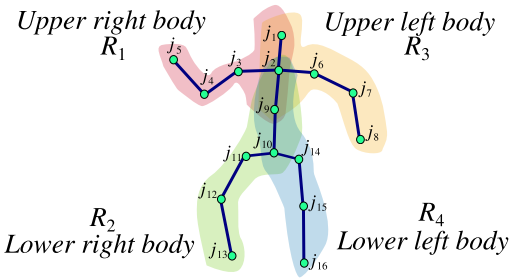

Figure 3. Skeleton representation used for splitting the human body into a set of spatial regions.

parts of the body that are involved in each action (spatial localization) along with the temporal span of each action (temporal localization). As our focus is on concurrent and composable activities, we would also like to encode multiple levels of abstraction, such that we can reason about poses, actions, and their compositions. Therefore, we develop a hierarchical compositional framework for modeling and recognizing complex human actions.

One of the key contributions of our model is its capability to spatially localize the body regions that are involved in the execution of each action, *both at training and testing time*. Our training process does not require careful spatial annotation and localization of actions in the training set; instead, it uses temporal annotations of actions only. At test time, it can discover the spatial and temporal span, as well as, the specific configuration of the main body regions executing each action. We now introduce the components of our model and the training process that achieves this goal.

3.1. Body regions

We divide the body pose into R fixed spatial regions and independently compute a pose feature vector for each region. Figure 3 illustrates the case when $R = 4$ that we use in all our experiments. Our body pose feature vector consists of the concatenation of two descriptors. At frame t and region r, a descriptor $x^g_{t,r}$ encodes geometric information about the spatial configuration of body joints, and a descriptor $x^m_{t,r}$ encodes local motion information around each body joint position. We use the geometric descriptor from [18]: we construct six segments that connect pairs of joints at each region[1] and compute 15 angles between those segments. Also, three angles are calculated between a plane formed by three segments[2] and the remaining three non-coplanar segments, totalizing an 18-D geometric descriptor (GEO) for every region. Our motion descriptor is based on tracking motion trajectories of key points [31], which in our case coincide with body joint positions. We extract a HOF descriptor using 32x32 RGB patches centered at the joint location for a temporal window of 15 frames. At each joint location, this produces a 108-D descriptor, which we concatenate across all joints in each a region to obtain our motion descriptor. Finally, we apply PCA to reduce the dimensionality of our concatenated motion descriptor to 20. The final descriptor is the concatenation of the geometric and motion descriptors, $x_{t,r} = [x^g_{t,r}; x^m_{t,r}]$.

3.2. Hierarchical compositional model

We propose a hierarchical compositional model that spans three semantic levels. Figure 2 shows a schematic of our model. At the top level, our model assumes that each input video has a single complex action label y. Each complex action is composed of a temporal and spatial arrangement of atomic actions with labels $\mathbf{u} = [u_1, \ldots, u_T]$, $u_i \in \{1, \ldots, S\}$. In turn, each atomic action consists of several non-shared *actionlets*, which correspond to representative sets of pose configurations for action identification, modeling the multimodality of each atomic action. We capture actionlet assignments in $\mathbf{v} = [v_1, \ldots, v_T]$, $v_i \in \{1, \ldots, A\}$. Each actionlet index v_i corresponds to a unique and known actomic action label u_i, so they are related by a mapping $\mathbf{u} = \mathbf{u}(\mathbf{v})$. At the intermediate level, our model assumes that each actionlet is composed of a temporal arrangement of a subset from K body poses, encoded in $\mathbf{z} = [z_1, \ldots, z_T]$, $z_i \in \{1, \ldots, K\}$, where K is a hyperparameter of the model. These subsets capture pose geometry and local motion, so we call them *motion poselets*. Finally, at the bottom level, our model identifies motion poselets using a bank of linear classifiers that are applied to the incoming frame descriptors.

We build each layer of our hierarchical model on top of BoW representations of labels. To this end, at the bottom

[1] Arm segments: wrist-elbow, elbow-shoulder, shoulder-neck, wrist-shoulder, wrist-head, and neck-torso; Leg segments: ankle-knee, knee-hip, hip-hip center, ankle-hip, ankle-torso and hip center-torso

[2] Arm plane: shoulder-elbow-wrist; Leg plane: hip-knee-ankle

level of our hierarchy, and for each body region, we learn a dictionary of motion poselets. Similarly, at the mid-level of our hierarchy, we learn a dictionary of actionlets, using the BoW representation of motion poselets as inputs. At each of these levels, spatio-temporal activations of the respective dictionary words are used to obtain the corresponding histogram encoding the BoW representation. The next two sections provide details on the process to represent and learn the dictionaries of motion poselets and actionlets. Here we discuss our integrated hierarchical model.

We formulate our hierarchical model using an energy function. Given a video of T frames corresponding to complex action y encoded by descriptors $\mathbf{x}$, with the label vectors $\mathbf{z}$ for motion poselets, $\mathbf{v}$ for actionlets and $\mathbf{u}$ for atomic actions, we define an energy function for a video as:

$$\begin{aligned} E(\mathbf{x},\mathbf{v},\mathbf{z},y) &= E_{\text{motion poselets}}(\mathbf{z},\mathbf{x}) \\ &+ E_{\text{motion poselets BoW}}(\mathbf{v},\mathbf{z}) + E_{\text{atomic actions BoW}}(\mathbf{u}(\mathbf{v}),y) \\ &+ E_{\text{motion poselets transition}}(\mathbf{z}) + E_{\text{actionlets transition}}(\mathbf{v}). \end{aligned} \quad (1)$$

Besides the BoW representations and motion poselet classifiers described above, Equation (1) includes two energy potentials that encode information related to temporal transitions between pairs of motion poselets ($E_{\text{motion poselets transition}}$) and actionlets ($E_{\text{actionlets transition}}$). The energy potentials are given by:

$$E_{\text{mot. poselet}}(\mathbf{z},\mathbf{x}) = \sum_{r,t}\left[\sum_{k} {w_k^r}^\top x_{t,r}\delta^k_{z_{(t,r)}} + \theta^r \delta^{K+1}_{z_{(t,r)}}\right] \quad (2)$$

$$E_{\text{mot. poselet BoW}}(\mathbf{v},\mathbf{z}) = \sum_{r,a,k} \beta^r_{a,k}\delta^a_{v_{(t,r)}}\delta^k_{z_{(t,r)}} \quad (3)$$

$$E_{\text{atomic act. BoW}}(\mathbf{u}(\mathbf{v}),y) = \sum_{r,s} \alpha^r_{y,s}\delta^s_{u(v_{(t,r)})} \quad (4)$$

$$E_{\text{mot. pos. trans.}}(\mathbf{z}) = \sum_{r,k_{+1},k'_{+1}} \eta^r_{k,k'}\sum_t \delta^k_{z_{(t-1,r)}}\delta^{k'}_{z_{(t,r)}} \quad (5)$$

$$E_{\text{acttionlet trans.}}(\mathbf{v}) = \sum_{r,a,a'} \gamma^r_{a,a'}\sum_t \delta^a_{v_{(t-1,r)}}\delta^{a'}_{v_{(t,r)}} \quad (6)$$

Our goal is to maximize $E(\mathbf{x},\mathbf{v},\mathbf{z},y)$, and obtain the spatial and temporal arrangement of motion poselets $\mathbf{z}$ and actionlets $\mathbf{v}$, as well as, the underlying complex action y.

In the previous equations, we use δ^b_a to indicate the Kronecker delta function $\delta(a=b)$, and use indexes $k \in \{1,\dots,K\}$ for motion poselets, $a \in \{1,\dots,A\}$ for actionlets, and $s \in \{1,\dots,S\}$ for atomic actions. In the energy term for motion poselets, w^r_k are a set of K linear pose classifiers applied to frame descriptors $x_{t,r}$, according to the label of the latent variable $z_{t,r}$. Note that there is a special label $K+1$; the role of this label will be explained in Section 3.5. In the energy potential associated to the BoW representation for motion poselets, β^r denotes a set of A mid-level classifiers, whose inputs are histograms of motion poselet labels at those frame annotated as actionlet a. At the highest level, α^r_y is a linear classifier associated with complex action y, whose input is the histogram of atomic action labels, which are related to actionlet assignments by the mapping function $\mathbf{u}(\mathbf{v})$. Note that all classifiers and labels here correspond to a single region r. We add the contributions of all regions to compute the global energy of the video. The transition terms act as linear classifiers η^r and γ^r over histograms of temporal transitions of motion poselets and temporal transitions of actionlets respectively. As we have a special label $K+1$ for motion poselets, the summation index k_{+1} indicates the interval $[1,\dots,K+1]$.

3.3. Learning motion poselets

In our model, motion poselets are learned by treating them as latent variables during training. Before training, we fix the number of motion poselets per region to K. In every region r, we learn an independent set of pose classifiers $\{w^r_k\}^K_{k=1}$, initializing the motion poselet labels using the k-means algorithm. We learn pose classifiers, actionlets and complex actions classifiers jointly, allowing the model to discover discriminative motion poselets useful to detect and recognize complex actions. As shown in previous work, jointly learning linear classifiers to identify body parts and atomic actions improves recognition rates [18, 34], so here we follow a similar hierarchical approach, and integrate learning of motion poselets with the learning of actionlets.

3.4. Learning actionlets

A single linear classifier does not offer enough flexibility to identify atomic actions that exhibit high visual variability. As an example, the atomic action "open" can be associated with "opening a can" or "opening a book", displaying high variability in action execution. Consequently, we augment our hierarchical model including multiple classifiers to identify different modes of action execution.

Inspired by [23], we use the *Cattell's Scree test* to find a suitable number of actionlets to model each atomic action. Specifically, using the atomic action labels, we compute a descriptor for every video interval using normalized histograms of initial pose labels obtained with k-means. Then, for a particular atomic action s, we compute the eigenvalues $\lambda(s)$ of the affinity matrix of the atomic action descriptors, which is build using χ^2 distance. For each atomic action $s \in \{1,\dots,S\}$, we find the number of actionlets G_s as $G_s = \text{argmin}_i\, \lambda(s)^2_{i+1}/(\sum^i_{j=1}\lambda(s)_j) + c\cdot i$, with $c = 2\cdot 10^{-3}$. Finally, we cluster the descriptors from each atomic action s running k-means with $k = G_s$. This scheme generates a set of non-overlapping actionlets to model each single atomic action. In our experiments, we notice that the number of actionlets used to model each atomic action varies typically from 1 to 8.

To transfer the new labels to the model, we define $u(v)$ as a function that maps from actionlet label v to the corresponding atomic action label u. A dictionary of actionlets

provides a richer representation for actions, where several actionlets will map to a single atomic action. This behavior resembles a max-pooling operation, where at inference time we will choose the set of actionlets that best describe the performed actions in the video, keeping the semantics of the original atomic action labels.

3.5. A garbage collector for motion poselets

While poses are highly informative for action recognition, an input video might contain irrelevant or idle zones, where the underlying poses are noisy or non-discriminative to identify the actions being performed in the video. As a result, low-scoring motion poselets could degrade the pose classifiers during training, decreasing their performance. To deal with this problem, we include in our model a *garbage collector* mechanism for motion poselets. This mechanism operates by assigning all low-scoring motion poselets to the $(K+1)$-th pose dictionary entry. These collected poses are associated with a learned score lower than θ^r, as in Equation (2). Our experiments show that this mechanism leads to learning more discriminative motion poselet classifiers.

3.6. Learning

Initial actionlet labels. An important step in the training process is the initialization of latent variables. This is a challenging due to the lack of spatial supervision: at each time instance, the available atomic actions can be associated with any of the R body regions. We adopt the machinery of self-paced learning [15] to provide a suitable solution and formulate the association between actions and body regions as an optimization problem. We constrain this optimization using two structural restrictions: i) atomic actions intervals must not overlap in the same region, and ii) a labeled atomic action must be present at least in one region. We formulate the labeling process as a binary Integer Linear Programming (ILP) problem, where we define $b^m_{r,q}=1$ when action interval $q \in \{1,\dots,Q_m\}$ is active in region r of video m; and $b^m_{r,q}=0$ otherwise. Each action interval q is associated with a single atomic action. We assume that we have initial motion poselet labels $z_{t,r}$ in each frame and region. We describe the action interval q and region r using the histogram $h^m_{r,q}$ of motion poselet labels. We can find the correspondence between action intervals and regions using a formulation that resembles the operation of$k$-means, but using the structure of the problem to constraint the labels:

$$\begin{aligned} \text{P1)} \quad & \min_{b,\mu} \sum_{m=1}^{M}\sum_{r=1}^{R}\sum_{q=1}^{Q_m} b^m_{r,q} d(h^m_{r,q}-\mu^r_{a_q}) - \frac{1}{\lambda} b^m_{r,q} \\ \text{s.t.} \quad & \sum_{r=1}^{R} b^m_{r,q} \geq 1, \forall q, \forall m \\ & b^m_{r,q_1} + b^m_{r,q_2} \leq 1 \text{ if } q_1 \cap q_2 \neq \emptyset, \forall r, \forall m \\ & b^m_{r,q} \in \{0,1\}, \forall q, \forall r, \forall m \end{aligned} \tag{7}$$

with

$$d(h^m_{r,q}-\mu^r_{a_q}) = \sum_{k=1}^{K} (h^m_{r,q}[k]-\mu^r_{a_q}[k])^2/(h^m_{r,q}[k]+\mu^r_{a_q}[k]). \tag{8}$$

Here, $\mu^r_{a_q}$ are the means of the descriptors with action label a_q within region r. We solve P1 iteratively using a block coordinate descending scheme, alternating between solving $b^m_{r,q}$ with μ^r_a fixed, which has a trivial solution; and then fixing μ^r_a to solve $b^m_{r,q}$, relaxing P1 to solve a linear program. Note that the second term of the objective function in P1 resembles the objective function of *self-paced* learning [15], managing the balance between assigning a single region to every action or assigning all possible regions to the respective action interval.

Learning model parameters. We formulate learning the model parameters as a Latent Structural SVM problem [39], with latent variables for motion poselets $\mathbf{z}$ and actionlets $\mathbf{v}$. We find values for parameters in equations (2-6), slack variables ξ_i, motion poselet labels $\mathbf{z}_i$, and actionlet labels $\mathbf{v}_i$, by solving:

$$\min_{W,\xi_i,\ i=\{1,\dots,M\}} \frac{1}{2}||W||_2^2 + \frac{C}{M}\sum_{i=1}^{M}\xi_i, \tag{9}$$

where

$$W^\top = [\alpha^\top, \beta^\top, w^\top, \gamma^\top, \eta^\top, \theta^\top], \tag{10}$$

and

$$\begin{aligned} \xi_i = \max_{\mathbf{z},\mathbf{v},y} \{ & E(\mathbf{x}_i,\mathbf{z},\mathbf{v},y) + \Delta((y_i,\mathbf{v}_i),(y,\mathbf{v})) \\ & - \max_{\mathbf{z}_i} E(\mathbf{x}_i,\mathbf{z}_i,\mathbf{v}_i,y_i)\}, \quad i \in [1,...M]. \end{aligned} \tag{11}$$

In Equation (11), each slack variable ξ_i quantifies the error of the inferred labeling for video i. We solve Equation (9) iteratively using the CCCP algorithm [40], by solving for latent labels $\mathbf{z}_i$ and $\mathbf{v}_i$ given model parameters W, temporal atomic action annotations (when available), and labels of complex actions occurring in training videos (see Section 3.7). Then, we solve for W via 1-slack formulation using Cutting Plane algorithm [12].

The role of the loss function $\Delta((y_i,\mathbf{v}_i),(y,\mathbf{v}))$ is to penalize inference errors during training. If the true actionlet labels are known in advance, the loss function is the same as in [18] using the actionlets instead of atomic actions:

$$\Delta((y_i,\mathbf{v}_i),(y,\mathbf{v})) = \lambda_y(y_i \neq y) + \lambda_v \frac{1}{T}\sum_{t=1}^{T}\delta(v_{ti} \neq v_t), \tag{12}$$

where v_{ti} is the true actionlet label. If the spatial ordering of actionlets is unknown (hence the latent actionlet formulation), but the temporal composition is known, we can compute a list A_t of possible actionlets for frame t, and include that information on the loss function as

$$\Delta((y_i,\mathbf{v}_i),(y,\mathbf{v})) = \lambda_y(y_i \neq y) + \lambda_v \frac{1}{T}\sum_{t=1}^{T}\delta(v_t \notin A_t) \tag{13}$$

3.7. Inference

The input to the inference algorithm is a new video sequence with features $\mathbf{x}$. The task is to infer the best complex action label $\hat{y}$, and to produce the best labeling of actionlets $\hat{\mathbf{v}}$ and motion poselets $\hat{\mathbf{z}}$.

$$\hat{y}, \hat{\mathbf{v}}, \hat{\mathbf{z}} = \underset{y,\mathbf{v},\mathbf{z}}{\operatorname{argmax}} E(\mathbf{x}, \mathbf{v}, \mathbf{z}, y) \tag{14}$$

We can solve this by exhaustively enumerating all values of complex actions y, and solving for $\hat{\mathbf{v}}$ and $\hat{\mathbf{z}}$ using:

$$\begin{aligned} \hat{\mathbf{v}}, \hat{\mathbf{z}}|y \;=\; \underset{\mathbf{v},\mathbf{z}}{\operatorname{argmax}} \sum_{r=1}^{R} \sum_{t=1}^{T} \Big(& \alpha^r_{y,u(v(t,r))} + \beta^r_{v_{(t,r)},z_{(t,r)}} \\ & + {w^r_{z_{(t,r)}}}^{\top} x_{t,r} \delta(z_{(t,r)} \leq K) + \theta^r \delta^{K+1}_{z_{(t,r)}} \\ & + \gamma^r_{v_{(t-1,r)},v_{(t,r)}} + \eta^r_{z_{(t-1,r)},z_{(t,r)}} \Big). \end{aligned} \tag{15}$$

4. Experiments

Our experimental validation focuses on evaluating two properties of our model. First, we measure action classification accuracy on several action recognition benchmarks. Second, we measure the performance of our model to provide detailed information about atomic actions and body regions associated to the execution of a complex action.

We evaluate our method on four action recognition benchmarks: the MSR-Action3D dataset [17], Concurrent Actions dataset [35], Composable Activities Dataset [18], and sub-JHMDB [11]. Using cross-validation, we set $K = 100$ in Composable Activities and Concurrent Actions datasets, $K = 150$ in sub-JHMDB, and $K = 200$ in MSR-Action3D. In all datasets, we fix $\lambda_y = 100$ and $\lambda_u = 25$. The number of *actionlets* to model each atomic action is estimated using the method described in Section 3.4. The garbage collector (GC) label $(K + 1)$ is automatically assigned during inference according to the learned model parameters θ^r. We initialize the 20% most dissimilar frames to the $K + 1$ label. In practice, at test time, the number of frames labeled as $(K + 1)$ ranges from 14% in MSR-Action3D to 29% in sub-JHMDB.

Computation is fast during testing. In the Composable Activities dataset, our CPU implementation runs at 300 fps on a 32-core computer, while training time is 3 days, mostly due to the massive execution of the cutting plane algorithm. Using Dynamic Programming, complexity to estimate labels is linear with the number of frames T and quadratic with the number of actionlets A and motion poselets K. In practice, we filter out the majority of combinations of motion poses and actionlets in each frame, using the 400 best combinations of (k, a) according to the value of non-sequential terms in the dynamic program. Details are provided in the supplementary material.

Algorithm	Accuracy
Our model	93.0%
L. Tao *et al*. [26]	93.6%
C. Wang *et al*. [30]	90.2%
Vemulapalli *et al*. [28]	89.5%

Table 1. Recognition accuracy in the MSR-Action3D dataset.

4.1. Classification of Simple and Isolated Actions

As a first experiment, we evaluate the performance of our model on the task of simple and isolated human action recognition in the MSR-Action3D dataset [17]. Although our model is tailored at recognizing complex actions, this experiment verifies the performance of our model in the simpler scenario of isolated atomic action classification.

The MSR-Action3D dataset provides pre-trimmed depth videos and estimated body poses for isolated actors performing actions from 20 categories. We use 557 videos in a similar setup to [32], where videos from subjects 1, 3, 5, 7, 9 are used for training and the rest for testing. Table 1 shows that in this dataset our model achieves classification accuracies comparable to state-of-the-art methods.

4.2. Detection of Concurrent Actions

Our second experiment evaluates the performance of our model in a concurrent action recognition setting. In this scenario, the goal is to predict the temporal localization of actions that may occur concurrently in a long video. We evaluate this task on the Concurrent Actions dataset [35], which provides 61 RGBD videos and pose estimation data annotated with 12 action categories. We use a similar evaluation setup as proposed by the authors. We split the dataset into training and testing sets with a 50%-50% ratio. We evaluate performance by measuring precision-recall: a detected action is declared as a true positive if its temporal overlap with the ground truth action interval is larger than 60% of their union, or if the detected interval is completely covered by the ground truth annotation.

Our model is tailored at recognizing complex actions that are composed of atomic components. However, in this scenario, only atomic actions are provided and no compositions are explicitly defined. Therefore, we apply a simple preprocessing step: we cluster training videos into groups by comparing the occurrence of atomic actions within each video. The resulting groups are used as complex actions labels in the training videos of this dataset. At inference time, our model outputs a single labeling per video, which corresponds to the atomic action labeling that maximizes the energy of our model. Since there are no thresholds to adjust, our model produces the single precision-recall measurement reported in Table 2. Our model outperforms the state-of-the-art method in this dataset at that recall level.

Algorithm	Precision	Recall
Our full model	0.92	0.81
Wei et al. [35]	0.85	0.81

Table 2. Recognition accuracy in the Concurrent Actions dataset.

Algorithm	Accuracy
Base model + GC, GEO desc. only, spatial supervision	88.5%
Base model + GC, with spatial supervision	91.8%
Our full model, no spatial supervision (latent $\mathbf{v}$)	91.1%
Lillo *et al.* [18] (without GC)	85.7%
Cao et al. [4]	79.0%

Table 3. Recognition accuracy in the Composable Activities dataset.

4.3. Recognition of Composable Activities

In this experiment, we evaluate the performance of our model to recognize complex and composable human actions. In the evaluation, we use the Composable Activities dataset [18], which provides 693 videos of 14 subjects performing 16 activities. Each activity is a spatio-temporal composition of atomic actions. The dataset provides a total of 26 atomic actions that are shared across activities. We train our model using two levels of supervision during training: i) spatial annotations that map body regions to the execution of each action are made available ii) spatial supervision is not available, and therefore the labels $\mathbf{v}$ to assign spatial regions to actionlets are treated as latent variables.

Table 3 summarizes our results. We observe that under both training conditions, our model achieves comparable performance. This indicates that our weakly supervised model can recover some of the information that is missing while performing well at the activity categorization task. In spite of using less supervision at training time, our method outperforms state-of-the-art methodologies that are trained with full spatial supervision.

4.4. Action Recognition in RGB Videos

Our experiments so far have evaluated the performance of our model in the task of human action recognition in RGBD videos. In this experiment, we explore the use of our model in the problem of human action recognition in RGB videos. For this purpose, we use the sub-JHMDB dataset [11], which focuses on videos depicting 12 actions and where most of the actor body is visible in the image frames. In our validation, we use the 2D body pose configurations provided by the authors and compare against previous methods that also use them. Given that this dataset only includes 2D image coordinates for each body joint, we obtain the geometric descriptor by adding a depth coordinate with a value $z = d$ to joints corresponding to wrist and knees, $z = -d$ to elbows, and $z = 0$ to other joints, so we can compute angles between segments, using $d = 30$ fixed with cross-validation. We summarize the results in Table 4, which shows that our method outperforms alternative state-of-the-art techniques.

Algorithm	Accuracy
Our model	77.5%
Huang et al. [11]	75.6%
Chéron et al. [5]	72.5%

Table 4. Recognition accuracy in the sub-JHMDB dataset.

Videos	Annotation inferred	Precision	Recall
Testing set	Spatio-temporal, no GC	0.59	0.77
Testing set	Spatio-temporal	0.62	0.78
Training set	Spatial only	0.86	0.90
Training set	Spatio-temporal	0.67	0.85

Table 5. Atomic action annotation performances in the Composable Activities dataset. The results show that our model is able to recover spatio-temporal annotations both at training and testing time.

4.5. Spatio-temporal Annotation of Atomic Actions

In this experiment, we study the ability of our model to provide spatial and temporal annotations of relevant atomic actions. Table 5 summarizes our results. We report precision-recall rates for the spatio-temporal annotations predicted by our model in the testing videos (first and second rows). Notice that this is a very challenging task. The testing videos do no provide any label, and the model needs to predict both, the temporal extent of each action and the body regions associated with the execution of each action. Although the difficulty of the task, our model shows satisfactory results being able to infer suitable spatio-temporal annotations.

We also study the capability of the model to provide spatial and temporal annotations during training. In our first experiment, each video is provided with the temporal extent of each action, so the model only needs to infer the spatial annotations (third row in Table 5). In a second experiment, we do not provide any temporal or spatial annotation, but only the global action label of each video (fourth row in Table 5). In both experiments, we observe that the model is still able to infer suitable spatio-temporal annotations.

4.6. Effect of Model Components

In this experiment, we study the contribution of key components of the proposed model. First, using the sub-JHMDB dataset, we measure the impact of three components of our model: garbage collector for motion poselets (GC), multimodal modeling of actionlets, and use of latent variables to infer spatial annotation about body regions (latent $\mathbf{v}$). Table 6 summarizes our experimental results. Table 6 shows that the full version of our model achieves the best performance, with each of the components mentioned above contributing to the overall success of the method.

Second, using the Composable Activities dataset, we also analyze the contribution of the proposed self-paced learning scheme for initializing and training our model. We summarize our results in Table 7 by reporting action recognition accuracy under different initialization schemes:

Algorithm	Accuracy
Base model, GEO descriptor only	66.9%
Base Model	70.6%
Base Model + GC	72.7%
Base Model + Actionlets	75.3%
Our full model (Actionlets + GC + latent $\mathbf{v}$)	77.5%

Table 6. Analysis of contribution to recognition performance from each model component in the sub-JHMDB dataset.

Initialization Algorithm	Accuracy
Random	46.3%
Clustering	54.8%
Ours	91.1%
Ours, fully supervised	91.8%

Table 7. Results in Composable Activities dataset, with latent $\mathbf{v}$ and different initializations.

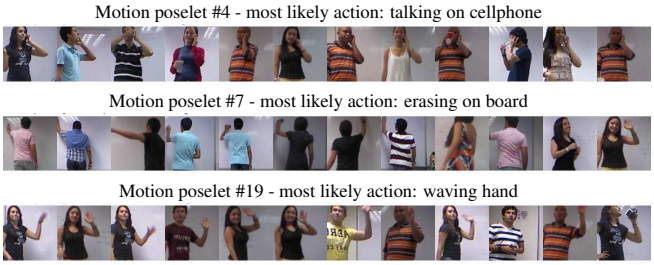

Figure 4. Moving poselets learned from the Composable Activities dataset.

i) Random: random initialization of latent variables $\mathbf{v}$, ii) Clustering: initialize $\mathbf{v}$ by first computing a BoW descriptor for the atomic action intervals and then perform k-means clustering, assigning the action intervals to the closer cluster center, and iii) Ours: initialize $\mathbf{v}$ using the proposed self-paced learning scheme. Our proposed initialization scheme helps the model to achieve its best performance.

4.7. Qualitative Results

Finally, we provide a qualitative analysis of relevant properties of our model. Figure 4 shows examples of moving poselets learned in the Composable Activities dataset. We observe that each moving poselet captures a salient body configuration that helps to discriminate among atomic actions. To further illustrate this, Figure 4 indicates the most likely underlying atomic action for each moving poselet. Figure 5 presents a similar analysis for moving poselets learned in the MSR-Action3D dataset.

We also visualize the action annotations produced by our model. Figure 6 (top) shows the action labels associated with each body part in a video from the Composable Activities dataset. Figure 6 (bottom) illustrates per-body part action annotations for a video in the Concurrent Actions dataset. These examples illustrate the capabilities of our model to correctly annotate the body parts that are involved in the execution of each action, in spite of not having that information during training.

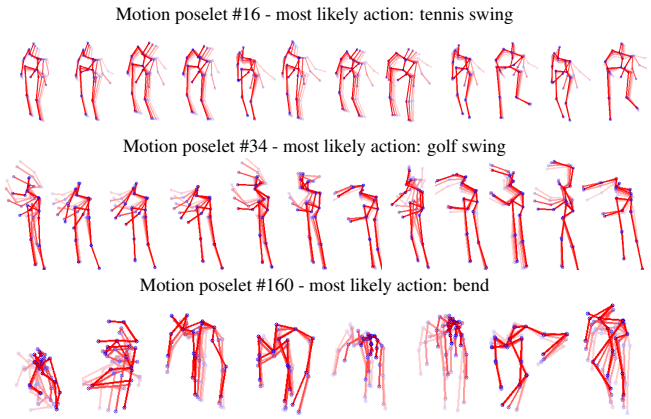

Figure 5. Moving poselets learned from the MSR-Action3D dataset.

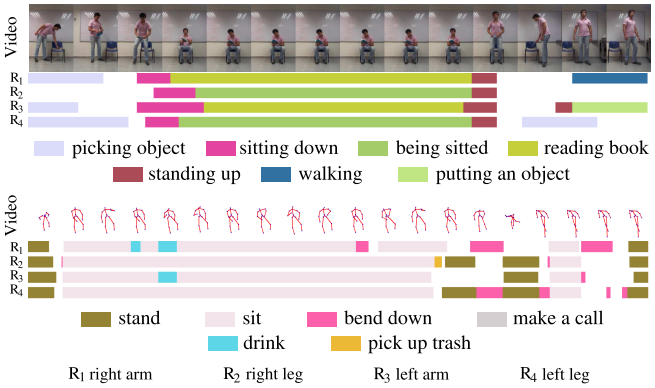

Figure 6. Automatic spatio-temporal annotation of atomic actions. Our method detects the temporal span and spatial body regions that are involved in the performance of atomic actions in videos.

5. Conclusions and Future Work

We present a hierarchical model for human action recognition using body joint locations. By using a semisupervised approach to jointly learn dictionaries of motions poselets and actionlets, the model demonstrates to be very flexible and informative, to handle visual variations and to provide spatio-temporal annotations of relevant atomic actions and active body part configurations. In particular, the model demonstrates to be competitive with respect to state-of-the-art approaches for complex action recognition, while also proving highly valuable additional information. As future work, the model can be extended to handle multiple actor situations, to use contextual information such as relevant objects, and to identify novel complex actions not present in the training set.

Acknowledgements This work was partially funded by the FONDECYT grant 1151018, from CONICYT, Government of Chile; and by the Stanford AI Lab-Toyota Center for Artificial Intelligence Research. I.L. is supported by a PhD studentship from CONICYT.

References

[1] J. K. Aggarwal and M. S. Ryoo. Human activity analysis. *ACM Computing Surveys*, 43(3):16:1–16:43, Apr. 2011. 1, 2

[2] L. Bourdev, S. Maji, T. Brox, and J. Malik. Detecting people using mutually consistent poselet activations. In *ECCV*, pages 168–181, 2010. 2

[3] L. Bourdev and J. Malik. Poselets: Body part detectors trained using 3D human pose annotations. In *ICCV*, pages 1365–1372, 2009. 2

[4] C. Cao, Y. Zhang, and H. Lu. Spatio-temporal triangular-chain crf for activity recognition. In *Proceedings of the 23rd Annual ACM Conference on Multimedia Conference*, pages 1151–1154. ACM, 2015. 7

[5] G. Chéron, I. Laptev, and C. Schmid. P-CNN: Pose-based CNN Features for Action Recognition. *ICCV*, pages 3218–3226, 2015. 2, 7

[6] Y. Du, W. Wang, and L. Wang. Hierarchical Recurrent Neural Network for Skeleton Based Action Recognition. In *CVPR*, pages 1110–1118, 2015. 2

[7] V. Escorcia, M. A. Davila, M. Golparvar-Fard, and J. C. Niebles. Automated vision-based recognition of construction worker actions for building interior construction operations using RGBD cameras. In *Construction Research Congress*, pages 879–888, 2012. 2

[8] X. Feng and P. Perona. Human action recognition by sequence of movelet codewords. In *3DPVT*, volume 16, pages 717–721. IEEE, 2002. 2

[9] J.-F. Hu, W.-S. Zheng, J. Lai, and J. Zhang. Jointly learning heterogeneous features for RGB-D activity recognition. In *CVPR*, pages 5344–5352, 2015. 2

[10] N. Hu, G. Englebienne, Z. Lou, and B. Krose. Learning latent structure for activity recognition. In *ICRA*, 2014. 2

[11] H. Jhuang, J. Gall, S. Zuffi, C. Schmid, and M. J. Black. Towards understanding action recognition. In *ICCV*, pages 3192–3199, 2013. 1, 6, 7

[12] T. Joachims, T. Finley, and C. Yu. Cutting-plane training of structural SVMs. *Machine Learning*, 77(1):27–59, 2009. 5

[13] G. Johansson. Visual perception of biological motion and a model for its analysis. *Perception & Psychophysics*, 14(2):201–211, 1973. 1

[14] Y. Kong and Y. Fu. Bilinear heterogeneous information machine for RGB-D action recognition. In *CVPR*, pages 1054–1062, 2015. 2

[15] M. P. Kumar, B. Packer, and D. Koller. Self-paced learning for latent variable models. In *NIPS*, pages 1189–1197, 2010. 2, 5

[16] T. Lan, Y. Zhu, A. R. Zamir, and S. Savarese. Action recognition by hierarchical mid-level action elements. In *ICCV*, pages 4552–4560, 2015. 2

[17] W. Li, Z. Zhang, and Z. Liu. Action recognition based on a bag of 3D points. In *CVPR*, pages 9–14, 2010. 6

[18] I. Lillo, A. Soto, and J. C. Niebles. Discriminative hierarchical modeling of spatio-temporally composable human activities. In *CVPR*, pages 812–819, 2014. 2, 3, 4, 5, 6, 7

[19] S. Maji, L. Bourdev, and J. Malik. Action recognition from a distributed representation of pose and appearance. In *CVPR*, pages 3177–3184, 2011. 2

[20] B. X. Nie, C. Xiong, and S.-c. Zhu. Joint action recognition and pose estimation from video. In *CVPR*, pages 1293–1301, 2015. 2

[21] R. Poppe. A survey on vision-based human action recognition. *Image and Vision Computing*, 28(6):976–990, 2010. 1, 2

[22] D. Ramanan and D. A. Forsyth. Automatic annotation of everyday movements. In *NIPS*, 2003. 2

[23] M. Raptis, I. Kokkinos, and S. Soatto. Discovering discriminative action parts from mid-level video representations. *CVPR*, pages 1242–1249, 2012. 4

[24] M. Rohrbach, S. Amin, M. Andriluka, and B. Schiele. A database for fine grained activity detection of cooking activities. In *CVPR*, pages 1194–1201, 2012. 2

[25] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from a single depth image. In *Communications of the ACM*, pages 116–124, 2011. 1, 2

[26] L. Tao and R. Vidal. Moving Poselets : A Discriminative and Interpretable Skeletal Motion Representation for Action Recognition. In *IEEE International Conference on Computer Vision Workshops*, pages 61–69, 2015. 2, 6

[27] C. Thurau and V. Hlavac. Pose primitive based human action recognition in videos or still images. In *CVPR*, pages 1–8, 2008. 2

[28] R. Vemulapalli, F. Arrate, and R. Chellappa. Human Action Recognition by Representing 3D Skeletons as Points in a Lie Group. In *CVPR*, pages 588–595, 2014. 2, 6

[29] S. Vishwakarma and A. Agrawal. A survey on activity recognition and behavior understanding in video surveillance. *The Visual Computer*, 29(10):983–1009, 2013. 1, 2

[30] C. Wang, Y. Wang, and A. L. Yuille. An approach to pose-based action recognition. In *CVPR*, pages 915–922, 2013. 1, 6

[31] H. Wang, A. Klaser, C. Schmid, and C.-L. Liu. Action recognition by dense trajectories. In *CVPR*, pages 3169–3176, 2011. 3

[32] J. Wang, Z. Liu, Y. Wu, and J. Yuan. Mining actionlet ensemble for action recognition with depth cameras. In *CVPR*, pages 1290–1297, 2012. 2, 6

[33] L. Wang, Y. Qiao, and X. Tang. Video Action Detection with Relational Dynamic-Poselets. In *ECCV*, pages 565–580, 2014. 2

[34] Y. Wang and G. Mori. Learning a discriminative hidden part model for human action recognition. In *NIPS*, pages 1721–1728, 2008. 2, 4

[35] P. Wei, N. Zheng, Y. Zhao, and S.-C. Zhu. Concurrent action detection with structural prediction. In *ICCV*, pages 3136–3143, 2013. 2, 6, 7

[36] D. Weinland, R. Ronfard, and E. Boyer. A survey of vision-based methods for action representation, segmentation and recognition. *Journal of Computer Vision and Image Understanding*, 115(2):224–241, 2011. 1, 2

[37] B. Yao and L. Fei-Fei. Modeling Mutual Context of Object and Human Pose in Human-Object Interaction Activities. In *CVPR*, pages 17–24. IEEE, 2010. 2

[38] S. Yeung, O. Russakovsky, N. Jin, M. Andriluka, G. Mori, and L. Fei-Fei. Every Moment Counts: Dense Detailed Labeling of Actions in Complex Videos. *arXiv:1507.05738*, 2015. 2

[39] C. Yu and T. Joachims. Learning structural svms with latent variables. In *ICML*, pages 1169–1176, 2009. 2, 5

[40] A. Yuille and A. Rangarajan. The concave-convex procedure. *Neural Computation*, 15(4):915–936, 2003. 5

[41] M. Zanfir, M. Leordeanu, and C. Sminchisescu. The Moving Pose: An Efficient 3D Kinematics Descriptor for Low-Latency Action Recognition and Detection. In *ICCV*, pages 2752–2759, 2013. 2