Adaptive RNN Tree for Large-Scale Human Action Recognition

Wenbo Li[1], Longyin Wen[2], Ming-Ching Chang[1], Ser-Nam Lim[2,3], Siwei Lyu[1]

[1]Computer Science Department, University at Albany, SUNY [2]GE Global Research [3]Avitas System, a GE Venture

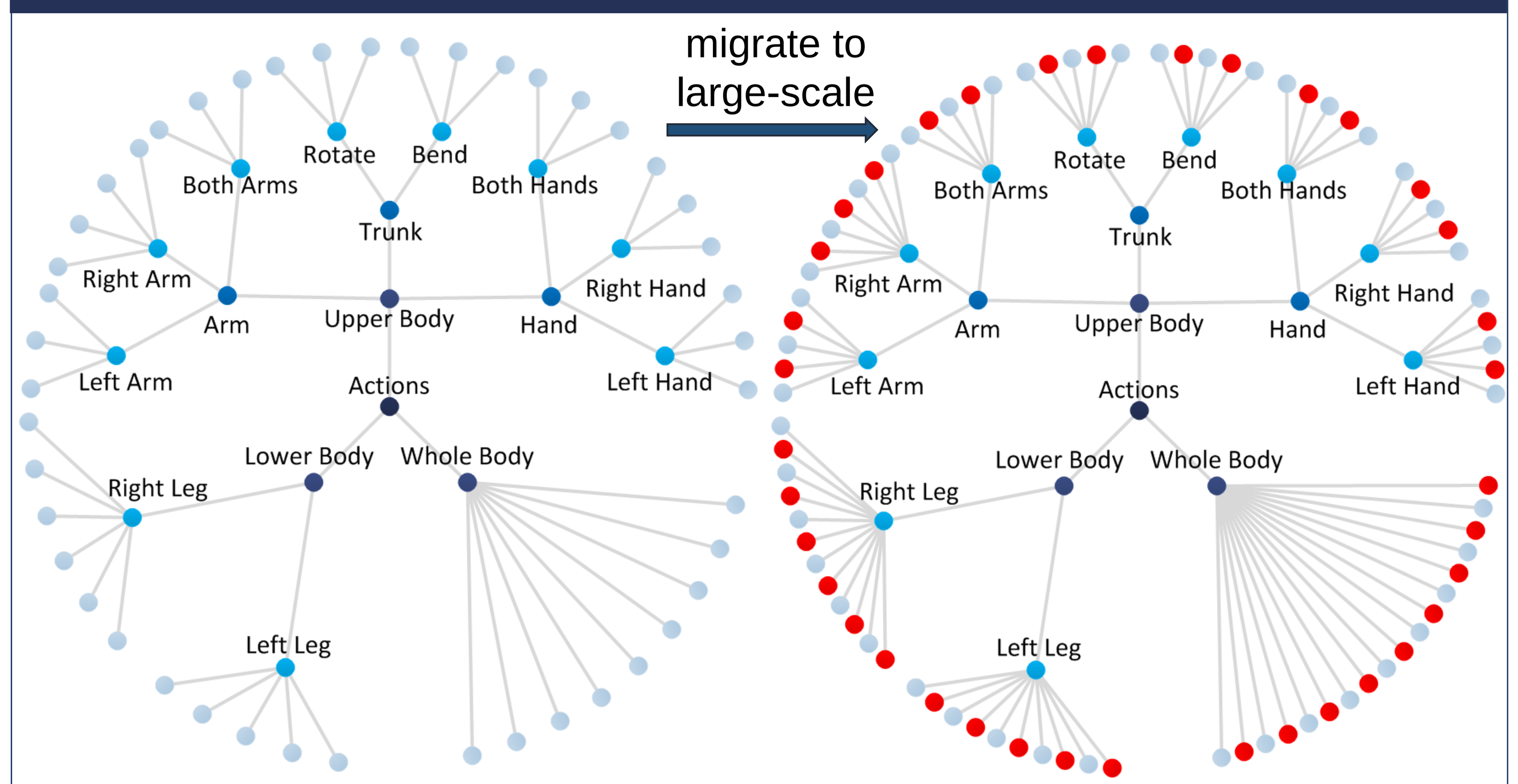
ABSTRACT

We present the RNN Tree (RNN-T), an adaptive learning framework for skeleton based human action recognition. Our method categorizes action classes and uses multiple Recurrent Neural Networks (RNNs) in a tree-like hierarchy. The RNNs in RNN-T are co-trained with the action category hierarchy, which determines the structure of RNN-T. Actions in skeletal representations are recognized via a hierarchical inference process, during which individual RNNs differentiate finer-grained action classes with increasing confidence. Inference in RNN-T ends when any RNN in the tree recognizes the action with high confidence, or a leaf node is reached. RNN-T effectively addresses two main challenges of large-scale action recognition:

(i) ***able to distinguish fine-grained action classes that are intractable using a single network***, and

(ii) ***(ii) adaptive to new action classes by augmenting an existing model.***

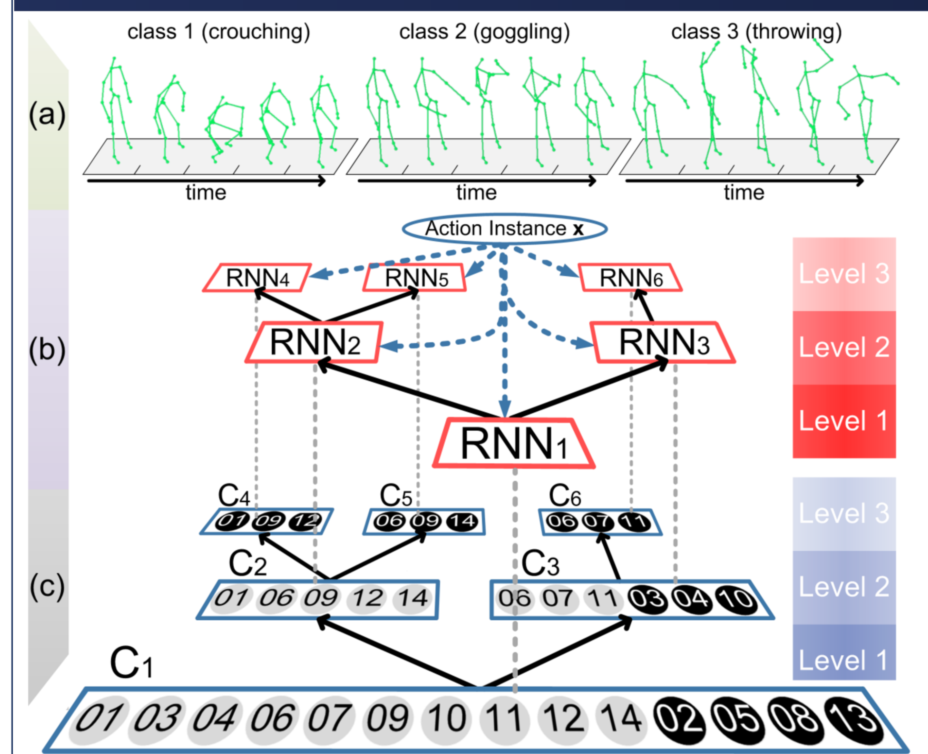
PROBLEM



CONTRIBUTIONS

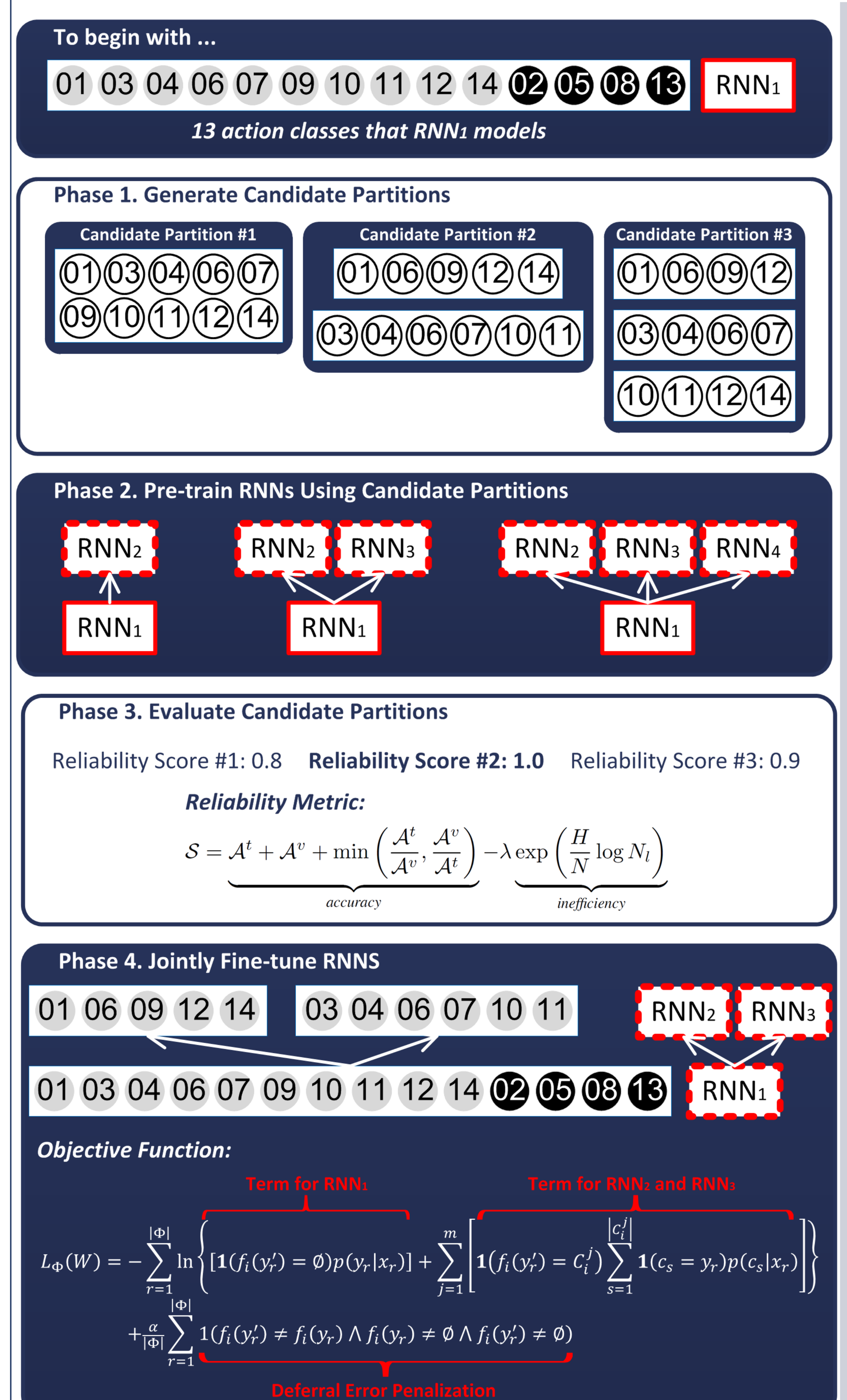
- Design an adaptive learning framework that aggregates multiple discriminative RNNs hierarchically for large-scale skeleton-based action recognition (SAR).
- Propose a novel, adaptive and hierarchical framework for fine-grained, large-scale SAR. Multiple RNNs are incorporated effectively in a tree-like hierarchy to mitigate the discriminative challenge using a divide-and-conquer strategy.
- Develop an effective learning procedure to build RNN-T to achieve high recognition accuracy and running efficiency.
- Design an incremental learning algorithm to make RNN-T adaptable to new classes and to significantly reduce the re-training time.
- Create a large-scale dataset, 3D-SAR-140, with the largest number of action classes to-date, and produce a benchmark to evaluate existing SAR methods and RNN-T based method.

OVERALL APPROACH



(a) Visualization of action instances from three action classes.
(b) A three-level RNN Tree (RNN-T) associated with the learned Action Category Hierarchy (ACH) in (c).
(c) Each circle represents an action class. Grey circles represent ambiguous classes, and black circles represent unambiguous ones. Action classes in the same box form one action category.

TRAINING

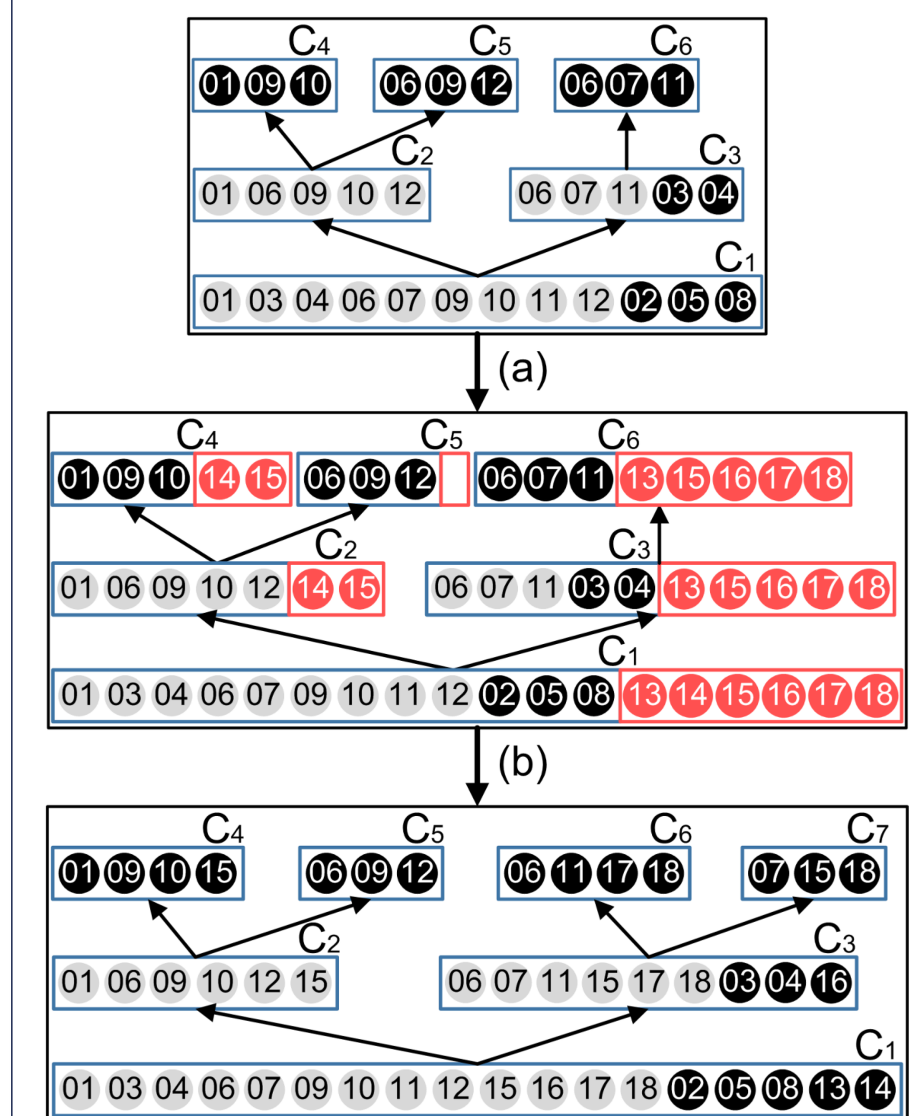


- **To begin with**, we have 13 classes, and train RNN_1 for them.
- **Phase 1.** We identify ambiguous classes, whose labels can not be confidently determined with RNN_1. These classes are divided into sub-categories to form new categories of the next level. Instead of using a fixed partition, we generate multiple candidate partitions by repeatedly running a spectral clustering algorithm.
- **Phase 2.** For each candidate partition, a set of RNNs are pre-trained independently.
- **Phase 3.** The optimal partition is determined based on a reliability metric, which captures the recognition accuracy for training and validation splits, and penalizes the inefficiency of the tree structure.
- **Phase 4.** RNNs corresponding to the newly generated categories are fine tuned jointly. $f_i(\cdot)$ is the lookup table of RNN*i* for deferral. C_i^j is the j-th child category of the i-th category. Φ represents the training dataset.

This material is based upon work partially supported (Siwei Lyu andWenbo Li) by the National Science Foundation under National Robotics Initiative Grant No. IIS-1537257.

GE Global Research

INCREMENTAL LEARNING



An example of ACH after each incremental learning procedure. Red circles represent new action classes.
(a) Insert new classes: All action categories accommodate new classes except C_5, which does not contain similar classes to the new ones.
(b) Update ACH and RNN-T: Minor changes occur in C_1, C_2, and C_4, and their corresponding RNNs are incrementally updated. The sub-tree starting from C_3 is rebuilt due to drastic changes.

EXPERIMENTS

We create a new dataset with 140 diverse action classes by aggregating all distinct classes from 10 existing datasets, which we name 3D-SAR-140. The 10 existing datasets are CMU Mocap [3] (23), ChaLearn Italian [6] (20), MSRC-12 Gesture [7] (12), MSR Action3D [16] (20), HDM05 [18] (65), Kintense [19] (10), Berkeley MHAD [20] (12), MSR Daily Activity 3D [29] (13), UTKinect-Action [32] (10), and ORGBD [34] (7), where the number of classes are shown in the parentheses. The number of sequences per class is 28 on average, and the frame rate is normalized to 20 frames-per-second (FPS), and the human skeleton is represented by 20 skeletal joints. We partition 60% of the 3D-SAR-140 as the training set, 20% as the validation set, and the remaining 20% as the testing set.

Table 1. Recognition results on 3D-SAR-140.

Methods	Accur.
URNN	0.296
URNN-L	0.665
BRNN	0.643
BRNN-L	0.672
URNN-2L	0.866
RR [28]	0.723
HBRNN-L [5]	0.604
CHARM [14]	0.618
DBN-HMM [31]	0.601
Lie-group [27]	0.745
HOD [8]	0.657
MP [35]	0.203
SSS [36]	0.253

Our Methods	Accur.
URNN-T	0.539
URNN-L-T	0.743
BRNN-T	0.705
BRNN-L-T	0.751
URNN-2L-T	**0.892**
HBRNN-L-T (4 levels)	0.756
HBRNN-L-T (3 levels)	0.750
HBRNN-L-T (2 levels)	0.735
HBRNN-L-T (1 level)	0.604
HBRNN-L-T w/o EJR	0.700
HBRNN-L-T w/o IP	0.697
HBRNN-L-T w/o FT	0.733

Figure 1. Incremental learning recognition results on 3D-SAR-140.

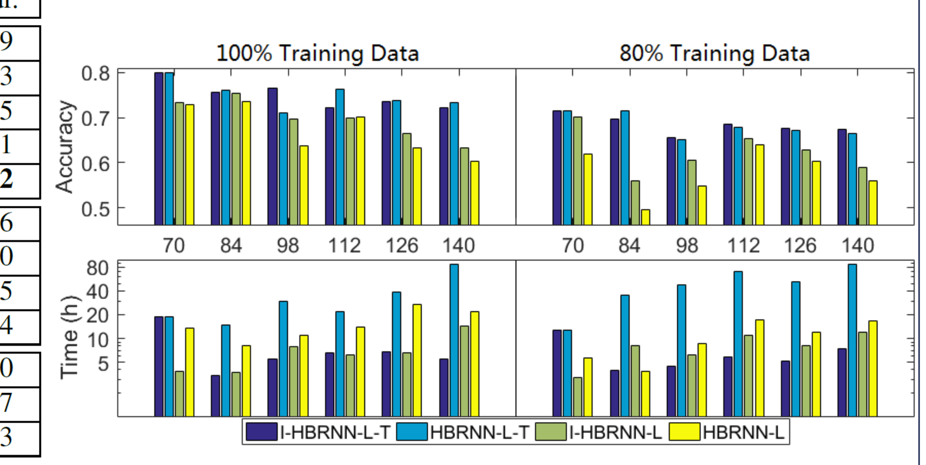


Figure 2. Recognition results on 10 existing datasets.