Pose and Joint-Aware Action Recognition

Anshul Shah[1] Shlok Mishra[2] Ankan Bansal[2] Jun-Cheng Chen[3]
Rama Chellappa[1] Abhinav Shrivastava[2]

[1]Johns Hopkins University [2]University of Maryland, College Park
[3]Research Center for Information Technology Innovation, Academia Sinica
{ashah95,rchella4}@jhu.edu {shlokm,ankan,abhinav2}@umd.edu pullpull@citi.sinica.edu.tw

Abstract

Recent progress on action recognition has mainly focused on RGB and optical flow features. In this paper, we approach the problem of joint-based action recognition. Unlike other modalities, constellation of joints and their motion generate models with succinct human motion information for activity recognition. We present a new model for joint-based action recognition, which first extracts motion features from each joint separately through a shared motion encoder before performing collective reasoning. Our joint selector module re-weights the joint information to select the most discriminative joints for the task. We also propose a novel joint-contrastive loss that pulls together groups of joint features which convey the same action. We strengthen the joint-based representations by using a geometry-aware data augmentation technique which jitters pose heatmaps while retaining the dynamics of the action. We show large improvements over the current state-of-the-art joint-based approaches on JHMDB, HMDB, Charades, AVA action recognition datasets. A late fusion with RGB and Flow-based approaches yields additional improvements. Our model also outperforms the existing baseline on Mimetics, a dataset with out-of-context actions.

1. Introduction

The task of action recognition has seen a lot of advances in recent times with improved spatio-temporal modeling, faster models and longer range temporal understanding. Most of the recent approaches in this area make use of raw RGB or dense optical flow as input features to reason about actions. In this paper, we approach the task of action recognition using joints and their motion. Unlike the commonly used dense optical flow, joints convey motion information succinctly, relying only on a sparse set of keypoints. Johansson's seminal work [20] on 'Moving Light Display' showed the importance of moving joints for human perception. In their experiments, bright spots were attached to joints of an actor dressed in black who was moving in front of a dark background. When the actor was not moving, the collection of spots was not helpful in discerning the action but when the person starts moving, the relative motion creates an impression of a person walking, dancing, etc. Further, many existing vision-based models tend to be biased by static context like scenes and objects [26, 27, 55]. Using joints helps to reduce these biases as it tends to be more robust to scene variations and explicitly captures only the human motion.

Recently, several interesting approaches for action recognition using human pose as a feature have been proposed. Approaches like PoTion[11], PA3D[59], SIP-Net[55] have shown impressive results. But, these approaches have certain limitations like overfitting on small datasets[11], requirement for access to multiple pose-modalities [59, 55], pose-tracking [55]. Many of these approaches also require access to features from a pose extractor, making it difficult for the model to be applied to different off-the-shelf pose extractors. Most importantly, we note that all of these approaches do a collective joint-reasoning step right from the first layer. Reasoning about activities in videos using joints requires the model to integrate motion information from multiple joints. But we found that early fusion of joint information does not let the model learn good per-joint motion signatures and might make the model rely a lot on co-occurrence patterns which can lead to sub-optimal representations.

In this paper, we propose to alleviate the aforementioned issues through our approach Joint-Motion Reasoning Network (JMRN). First, in order to learn richer motion representations, we extract per joint motion information using a shared motion encoder. This is followed by a collective joint reasoning module to infer the final activity. Our approach of separating per-joint feature extraction and collective reasoning allows for additional constraints on the mo-

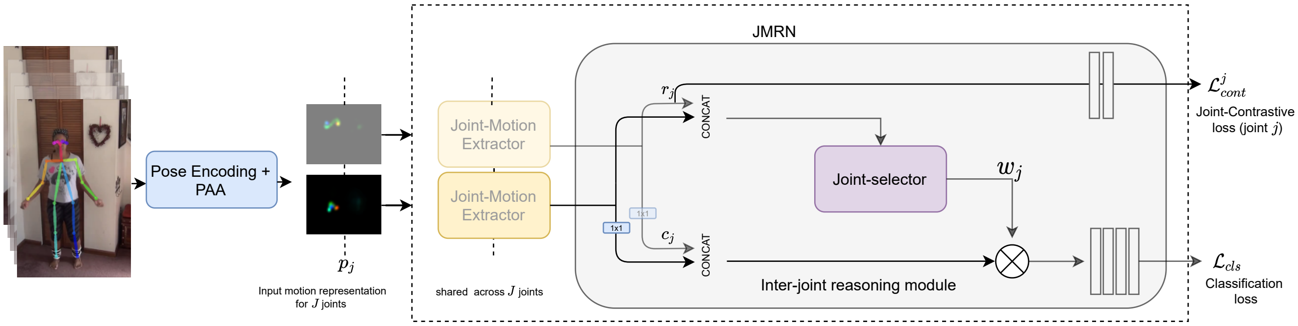

Figure 1: The overall pipeline. We propose the joint-motion re-weighting network (JMRN) to better capture inter-dependencies between joints. This model captures motion information while learning to use the most discriminative joints. We first encode joint trajectories for each joint and augment them with the Pose-Aware augmentation to obtain the input motion representations $p_j, \ \ j \in [1, \cdots, J]$. The joint-motion extractor module is a shared encoder that learns information about each of the J joints separately. This information is then forwarded to the inter-joint reasoning module, which learns a weight w_j for each joint and reweighs the representation. In addition to the classification loss, we enforce a novel contrastive loss on the MLP-projected joint representations. The joint-contrastive loss is calculated for each joint separately.

tion representations. We propose a novel joint-contrastive loss which pulls together features from the same action for each joint. This loss enforces that the per-joint motion signatures are expressive of the action. Different from the usual contrastive learning setup [7, 23] where the loss is applied to the last layer features, the proposed loss operates on the 'mid-level' joint features. To improve the generalization capability of the model, we propose to use two geometry-aware data augmentation techniques. Fig. 1 shows the schematic of our model. Our proposed approach is not tied to any specific pose-extractor and does not require any pose tracking, which is difficult for in-the-wild videos.

Using the proposed approach, we obtain an absolute accuracy improvement of 7.35% on JHMDB [19], 3.9% on HMDB[24], 2.4 mAP on Charades [41], and 1 mAP on AVA[17] over approaches which use pose heatmaps alone. When combined with recent RGB and flow-based methods, our method yields additional improvements without retraining the backbone, demonstrating its effectiveness. We also evaluate our method on the Mimetics [55] dataset. This dataset consists of out-of-context actions performed by mime artists. Evaluation on this dataset is meant to show the generalization capability of action recognition approaches. Standard RGB and flow-based models show dismal performance on this dataset. Our approach outperforms the baseline reported in [55] by 1.5% on top 1 accuracy without performing any tracking and only relying on joint heatmaps. Extensive analysis illustrates the importance of each proposed technique, and we often find that even our individual techniques outperform previous approaches. Implementation and preprocessed data is available[1].

In summary, we make the following contributions:

[1]https://github.com/anshulbshah/PoseAction

- We propose a new model to improve reasoning over multiple human joints. This model learns motion representations for each joint independently, then uses an inter-joint reasoning module to combine all the motion cues from various joints to make a prediction.
- Our novel joint-contrastive loss guides the joint representations to extract richer features. To the best of our knowledge, a contrastive learning loss on 'mid-level' joint features has not been explored in the past.
- We propose a geometry-aware data augmentation method to deal with limited data and improve generalization.
- We obtain state-of-the-art results on JHMDB, HMDB, Charades and AVA when using the joints alone. The late fusion of our approach with RGB and flow-based models gives improvements showing the complementary nature of pose information. Our model also obtains an improvement on Mimetics[55], a dataset with out-of-context actions using only pose heatmaps and without any tracking.

2. Related Works

Action Recognition from Videos. [42] was amongst the first to successfully apply deep networks for action recognition that relied on RGB and optical flow. [6] improved performance on action recognition task by inflating weights from 2D CNNs. Some recent advances in learning improved spatio-temporal representations [51, 40, 52, 62, 8, 56, 21, 15] have led to better performance in standard action recognition benchmarks. But, studies [26, 27, 55] show that RGB and flow-based models capture a lot of dataset biases. Unlike RGB and flow, human joints and their motion offer a succinct representation of the underlying activities and is less susceptible to dataset biases. To this end, we present an approach for action recognition using pose information.

Pose-based Action Classification. [19] showed that pose features perform much better than low/mid features and thus can act as discriminative cues for action recognition. Strides of improvements in pose estimation from images and videos [3, 30, 37, 47, 14, 46, 22, 5, 53, 61, 45, 33] have enabled extraction of pose information from in-the-wild videos for the task of action recognition.

In P-CNN [10], the authors extract appearance and flow features conditioned on the human pose. Similarly, chained multi-stream networks [63] combines the most important visual cues from pose, motion and RGB. A Multi-task deep learning framework was proposed in [31] for using visual information and joints to classify activities. A recurrent pose-attention network was introduced in [12] to learn the spatio-temporal features by attending to joints and extracting visual features. [2] also extracts visual features near joints to obtain video descriptors. However, handling multiple people becomes difficult with such a method. These methods require access to ground-truth keypoints during training, making it difficult to train on videos without these annotations. A rank pooling-based approach was proposed in [28] to pool information from the evolution of pose-heatmaps.

Most related to our method, [11] proposed to represent the pose evolution as a compact representation and used it in a deep framework to obtain improvements for action recognition in real-world videos. [59] improved upon this by using Part Affinity Fields [4] and visual features from the pose extraction pipeline along with a temporal pooling module. In contrast, we do not use these features and work on joint heatmaps alone, which is a common representation used in all pose extractors and shows improvements without using extra modalities. STAR-Net [32], reprojects pose-estimation features in space and time to build an end-to-end system for action recognition. While they use clips, we work with the entire video. Further, in addition to demonstrating better results, our approach is modular and can work with any pose estimation model. DynaMotion [1] proposed using a dynamics-based encoder which encodes input sequences using a structured dictionary. Similar to [11], they reason on all joints together. By extracting motion information from each joint independently before performing inter-joint reasoning helps our model extract richer motion representation which leads to superior results. This way of modeling also naturally allows us to use additional supervision on joint motion representations which leads to additional gains.

Another line of work exploits the skeletal structure of humans to classify activities [49, 54, 43, 60, 39]. Accurate 3D poses can be obtained using depth cameras, but it is not viable in general settings and obtaining 3D pose from unconstrained videos is still a very difficult problem. Hence we do not focus on these approaches in our work.

3. Method

In this section we first describe how we obtain the input per-joint motion trajectories for our approach, followed by our proposed approach

Pose extraction and encoding pose evolution In this paper, we focus on action recognition from videos using the evolution of human pose as a cue. Given a video of T frames, we first run a pose detector on each frame of the video to obtain heatmaps $h_j^t \in \mathbb{R}^{H \times W},\ t = 1, \ldots, T$ for the j^{th} joint, $j = 1, \ldots, J$. The heatmaps intuitively denote the location estimate of each joint for every frame. Following [11], we temporally aggregate the heatmaps using a weighted sum. Given the temporally stacked joint heatmaps $C \in \mathbb{R}^{T \times J \times H \times W}$, we aggregate the heatmaps to give a representation $P' = \sum_t C[t]o[t]$ using a (fixed) piece-wise linear weighing function o where $P' \in \mathbb{R}^{J \times H \times W}$. We use three different weighting functions described in [11], giving the final motion representation of shape $p_j \in \mathbb{R}^{3 \times H \times W}$. The advantage of using this approach is that it lets us encode the whole video without sampling, and also helps in reasoning about the global structure more effectively using CNNs. This module returns joint-motion trajectories which are used by our model for action recognition.

3.1. The Proposed Model and Training

Next, we introduce the structure of our JMRN which captures the joint correlations. Previous approaches used all joints concatenated together as input to the neural network. This can lead to the model not exploiting useful motion information from individual joints and relying on specific spatial arrangements of these found in the dataset. We argue that we can learn improved representations by independently extracting motion information from each joint using a shared module before combining the cues for inter-joint reasoning. Our next observation is that, for an activity, some joints might have more discriminative information than others. Some joints might act as 'distractors' to the training process and only provide redundant or noisy information. We can learn enhanced features by using information from more discriminative joints. Our model (Fig. 1), JMRN is designed to solve these issues and learns effective representations for joint-based action recognition. The model consists of two modules: The first module extracts information from all joints separately. The second module is an inter-joint reasoning module that is conditional on the input and learns to weigh the features from the various joints before reasoning over altogether. The joint-selector module is similar in spirit to Squeeze-and-Excitation block [18], but with a key difference that instead of operating on higher-level representations of the entire input, our representations contain motion information about that joint alone. This makes our selector module naturally interpretable. Finally, we stack the weighted motion representations and feed them to the inter-joint reasoning module. This module generates final logits by performing collective reasoning over all joints.

Joint-Motion Extraction. The joint-motion extraction module is a Siamese network that extracts information from each joint separately. The input to the network is the motion representation for the j^{th} joint, $p_j \in \mathbb{R}^{3\times H\times W}$ The module generates a representation $r_j \in \mathbb{R}^{256\times H\times W}$, $j = 1, \ldots, J$ which is then used by the joint-selector module. In addition, we also generate a compressed representation $c_j \in \mathbb{R}^{c_{dim}\times H\times W}$ which is obtained by passing r_j through a 1×1 convolution layer to reduce the number of channels.

Inter-Joint Reasoning. The joint-selector module pools information from all joint-motion representations r_j to obtain weights that are used to modulate the representations c_j. Specifically, we concatenate all the joint representations and apply a 1×1 convolution. This is followed by an average pooling operation which generates a feature of dimension 256. A linear layer followed by sigmoid activation is used to generate the weights w_j . Finally, we weigh the compressed representations c_j by the obtained weights of w_j and concatenate them. We use a 1×1 convolution to reduce the large dimension of the input ($J \times c_{dim}$) and follow this with two convolutional layers and an FC layer to give the final class logits.

Joint-Contrastive Loss The classification loss will naturally try to cluster final feature representations from the same class together. Extracting per-joint features before collective joint reasoning allows enforcing additional constraints on the joint features. Consider an instance of a person 'running'. While the standard classification loss enforces that concatenated features from instances of running lie close to each other, we can learn improved motion features by enforcing consistencies at the joint level. Specifically, our joint-contrastive loss ensures that per-joint features for instances of 'running' lie closer to each other than per-joint features of 'push-up'. We enforce this constraint by employing contrastive learning loss [35, 7, 23]. Different from prior works, our loss operates on 'mid-level' joint motion features. Further, we apply this loss for each joint separately. Our positives come from augmented examples of the same instance and other instances of the same activity while the negatives are other remaining instances from the batch. We first project the joint features r_j through an MLP and normalize them to obtain features $\boldsymbol{z_j}$ which lie on a unit hypersphere. Let the corresponding label for the instance be denoted by $\boldsymbol{y}$. The joint-contrastive loss is then defined by:

$$\mathcal{L}_{cont} = \sum_{j=1}^{j=J} \mathcal{L}^j_{cont} \tag{1}$$

$$\mathcal{L}^j_{cont} = \sum_{i\in B} \frac{-1}{|P(i)|} \sum_{p\in P(i)} \log \frac{\exp\left(\boldsymbol{z}^i_j \boldsymbol{z}^p_j/\tau\right)}{\sum\limits_{a\in B\setminus i} \exp\left(\boldsymbol{z}^i_j \boldsymbol{z}^a_j/\tau\right)} \tag{2}$$

where B are the instances in the batch, z^i_j denotes the projected features for the joint j of instance i and $P(i) \equiv \{p \in B \setminus i : \boldsymbol{y}^p = \boldsymbol{y}^i\}$, is the set of all instances which share the same label as instance i (positives). τ is the temperature parameter. To the best of our knowledge, a contrastive learning loss on joint features to improve representations has not been explored in the past.

3.2. Pose-Aware Data Augmentation (PAA)

Data augmentation, like flipping the representation, has been shown to be useful in previous approaches [11, 59]. We propose a data augmentation strategy which can use the geometric structure of the human joints. Our pose-aware data augmentation is a cascade of two operations. The first operation is a random global jitter of the entire input representation. This helps the network to learn that the action is not strongly dependent on the global position of joint trajectories and ensures that the relative position among the different joints is not modified. Another benefit is that it regularizes training by sometimes pushing joints out of the frame which emulates joints not being visible in the scene. In the second operation, we divide the pose into six groups for joints corresponding to the Head: {Nose, REye, LEye, LEar, REar}, Torso: {RHip, LHip, Neck}, Left Hand: {LShoulder, LElbow, LWrist}, Right Hand: {RShoulder, RElbow, RWrist}, Left Leg: {LKnee, LAnkle}, and Right Leg: {RKnee, RAnkle}. Each joint inside a group is randomly jittered by the same amount. As our first step involves a global motion, we do not jitter the joints in the torso group. This strategy amounts to adding random spatial noise but is pose-aware in the sense that all joints do not move by a different random amount. For example, the motion representations corresponding to eyes, ears and nose are jittered by the same amount. We represent our augmentation by parameters β, γ, which represents the maximum amount of random jitter for each of the two strategies, respectively. The same amount of jitter is used for all frames of a video to preserve the geometric integrity of the action. Similar to pose-aware translation, we also experiment with pose-aware rotation which performs comparably. We include those experiments in the supplementary material. We note that while some of these data augmentation techniques have been explored for pose-estimation, these have not been investigated for the task of action recognition which is a very different task conceptually and might require the model to learn different kind of invariances.

4. Experiments

In this section, we present experimental results to show the effectiveness of the proposed approach. Following prior works, we use HMDB [24], JHMDB [19], Charades [41], AVA[17] and Mimetics [55] datasets for our experiments.

HMDB [24] consists of 6,766 video clips from fifty-one action classes. Each video clip is trimmed and corresponds to a single action. There are 3,570 training videos and 1,530 videos for validation in each of the three splits.

JHMDB [19] is a subset of HMDB that contains 928 short videos from twenty-one action classes. The dataset comprises of $\sim$660 training videos and $\sim$268 validation videos for each of the three splits.

Charades [41] is a more recent dataset that contains daily-life videos instead of videos from movies or Youtube. The dataset has 157 action classes with a total of 9,848 untrimmed videos, of which 7,981 videos are for training and 1,863 are for validation.

AVA Actions [17] is a large-scale dataset which evaluates video recognition models on spatio-temporal action localization. One frame every second is annotated with actors and each actor might be involved in multiple actions. We use the AVA v2.1 set which consists of 211k training segments and 57k validation segments. Following previous works, we evaluate mAP at frame-level IoU threshold of 0.5 on sixty actions which have at least twenty-five instances in the validation set.

Mimetics [55] is a test set for fifty actions from the Kinetics dataset. It consists of 713 videos collected from Youtube that have mimed actions. Due to the small size of this dataset, it is only used for testing a model that is trained with Kinetics or its subset. Unlike most common action recognition datasets, this dataset consists of minimal scene and object biases and hence can be used to evaluate a method for out-of-context actions.

4.1. Implementation Details

For all the datasets we extract pose using an off-the-shelf pose detector [3] which gives heatmaps per frame for the video. We use the COCO trained pose extractor which returns 18 joints + background. We train the models using the Adam optimizer with a learning rate drop on accuracy plateau. We used cross-entropy for single label classification tasks and binary cross-entropy for multi-label tasks. Models are trained with the loss function $\mathcal{L}_{cls} + \lambda \mathcal{L}_{cont}$ where λ is chosen empirically. For multi-label tasks, two instances are considered positives if they share any of the labels. We used our own implementation of PoTion (baseline) since the official implementation is not publicly available.

4.2. Results

We show the results of our proposed approaches in Table 1. The baseline model is PoTion [11]. The baseline model stacks heatmaps from all joints and uses the combined representation to recognize actions. The proposed model, JMRN first extracts motion information from each joint using a shared module before reweighing this information for inter-joint reasoning. This gives consistent gains on all datasets. Adding the novel joint-contrastive loss to enforce feature consistency at the joint-level brings in additional improvements. Our proposed pose-aware data augmentation gives a considerable improvement to the baseline model and our JMRN model over all datasets.

Comparison with prior Pose-based Methods. In Tables 2 and 3, we compare our approach to other state-of-the-art approaches for the various datasets. The proposed approach performs better than other approaches that use only joint heatmaps. Our approach is more general, as unlike PAFs and CNN features, heatmaps can be obtained from any pose extractor. Further, unlike our method, PA3D's model uses additional inputs (PAF, CNN features, and temporal difference of features) and make use of ensembles of models. Our superior results without any of these bells and whistles show the benefit of our approach.

Fusion with State-of-the-Art Approaches. While joint motion is a strong cue for human activities, many of the current datasets have activities that are strongly dependent on objects and scenes. Our pose stream can still offer complementary motion information for such datasets and lead to an improved reasoning of activities. Naïvely averaging logits is sub-optimal for fusion since the logits are generated by models which are trained separately and might have a different effective range. We use a very simple learnt fusion scheme to combine the different modalities. Specifically, we learn a single scalar weighing parameter for each of the M modalities to be fused and M for each class for multi-label problems. This approach does not require back-propagation through the backbone and hence can be used as a quick post-training step with extracted logits. Fusing the scores with the best model, we obtain improvements on all datasets as shown in Tables 4 and 3. These results show that our model extracts joint information which is complementary to RGB and Flow. For HMDB and JHMDB, we fuse with I3D [6], and additionally ResNeXt101-BERT [21] which is the current state-of-the-art for HMDB. For Charades, we perform fusion with Long-Term Feature banks (LFB) [56]. We see that our model enables complementary gains throughout. We found that on HMDB-1 test set, fusion of JMRN over ResNeXT101-BERT[21] was particularly useful for `jump` (13%), `turn` (+7%), `draw-sword` (+7%) and `run` (+7%). On Charades, we reap the largest absolute improvements on `standing up` (+11%), `sitting down` (+10%) and `washing a mirror` (+10%) over the LFB model. These actions are strongly tied to human motion and the improvements justify the use of a pose stream in addition to RGB and optical

Table 1: Improvements using the proposed approach. We compare our model agains the baseline (PoTion[11]). Our proposed model outperforms the baseline by a large margin. Use of the proposed data augmentation step (PAA) improves the JMRN model consistently across the three datasets. Finally, using the novel joint-contrastive loss in addition to the classification loss gives further improvements.

	Approach	JHMDB-1	HMDB-1	Charades
	Baseline (PoTion)	59.44	42.04	13.54
Proposed Model	JMRN	66.70	48.71	15.00
	+ PAA	69.81	52.02	15.79
	+ PAA + Joint-Contrastive Loss	**71.08**	**54.05**	**16.2**

Table 2: Comparison with state-of-the-art on J-HMDB and HMDB datasets. We report mean-per class accuracy averaged over 3 splits. We obtain an improvement of +7.35% on JHMDB and +3.9% on HMDB when using pose heatmaps alone. Note that the best results for [59] make use of extra modalities and additional networks to process these.

Method	Features Used	HMDB	JHMDB
PA3D [59]	P + CF + PAF	55.3*	69.5*
Potion [11]	P	43.7	57.0
PA3D [59]	P	47.8	60.1
PA3D [59]	P	50.3*	61.2*
STAR-Net [32]	CF	-	64.3
EHPI [29]	P	-	60.5
DynaMotion [1]	P	49.1+	60.2+
SIP-Net [55]	CF	51.2	62.4
Ours	P	**54.2**	**68.55**

Models with * use difference features along with standard heatmaps and use an ensemble pose model. + are results on first split alone. P - pose heatmaps, PAF - Part affinity Fields [4], CF - Features from the pose extractor CNN.

Table 3: Classification mAP on the evaluation set of Charades dataset. We obtain an absolute improvement of 1.55 mAP over PA3D which uses PAF [4] and convolutional features apart from heatmaps. We also show results on combination with RGB (R) and Flow-based (F) approaches which gives further gains.

Method	Features	mAP
2Stream [42]	R + F	11.9
Asyn-TF [40]	R + F	22.4
I3D [6]	R	32.72†
GCN [50]	R	36.2
NL I3D [52]	R	37.5
R50-I3D-NL [56]	R	38.29#
R101-I3D-NL LFB [56]	R	42.5#
PoTion [11]	P	13.54†
Potion + R101-NL-LFB	P + R	42.84
PA3D [59]	P + CF + PAF	13.8
PA3D [59] + GCN + I3D + NL-I3D	P + CF + PAF + R	41.0
Ours	P	**16.2**
Ours + R101-NL-LFB	P + R	**43.23**

† denotes results we have reproduced. # using official implementation. P - pose heatmaps, PAF - Part affinity Fields [4], CF - Features from the pose extractor CNN.

flow. We believe that jointly training these models would lead to further improvements wherein each stream can specialize for different classes, but we leave that to future work.

Results on AVA Actions [17] To show the advantage of using a pose-stream on large-scale datasets, we perform experiments on the AVA Actions dataset. We compare our approach with the best publicly available SlowFast [15] model for this split. For fair comparison, we use bounding boxes provided by SlowFast [15]. Pose is extracted for each frame in the clip and bounding box information is used to crop per-person pose information across the clip. These are then used to obtain the pose encoding. Since some actions involve multi-person context information, we also append the pose encoding of other people in the clip as separate channels. Our results are shown in Table 5. We obtain an improvement over the strong RGB-based baseline using late fusion with JMRN's pose stream. It is particularly noteworthy that, unlike the SlowFast model which uses Kinetics-600 to pretrain the model, we train our model from scratch and our pose-only model has a comparable performance to the I3D RGB model.

Results on Mimetics dataset Mimetics [55] introduced a test set for a subset of fifty actions found in the Kinetics dataset. The proposed dataset is collected from Youtube but consists of mimed actions and can be used to evaluate the performance of models on out-of-context actions and thus evaluate their generalization capability. Pose can be a strong cue in such cases as the videos have minimal scene and object biases evident from the observation that RGB/Flow-based models have poor performance on this dataset [55].

Table 4: Comparison with fusion over recent state-of-the-art on HMDB and JHMDB dataset. We report mean-per class accuracy averaged over 3 splits. The improvement through model fusion shows that our approach is complementary to RGB (R) and flow-based (F) models.

Method	**Modality**	**HMDB**
2Stream [42]	R + F	59.4
TSN [51]	R + F	69.4
S3D [57]	R + F	75.9
R(2+1)D [48]	R + F	78.7
I3D [6]	R + F	81.09†
EvaNet [38]	R + F	82.3
PA3D [59] + I3D [6]	P + PAF + CF + R + F	82.1*
ResNext101 BERT [21]	R + F	83.76†
Ours + I3D	P + R + F	82.33
Ours + ResNext101 BERT	P + R + F	**84.53**

Method	**Modality**	**JHMDB**
I3D [6]	R + F	86.8†
P-CNN [10]	R + F	61.1
P-CNN + IDT [10]	R + F	71.4
Action Tubes [16]	R + F	62.5
MR-TS R-CNN [36]	R + F	71.1
GRP + IDT [8]	R + F	74.3
KRP + IDT [9]	R + F	74.2
Chained MultiStream [63]	P + R + F	76.1
PoTion[59] + I3D [6]	P + R + F	85.5
PA3D[59] + RPAN [13]	P + PAF + CF + R + F	86.1
Ours + I3D	P + R + F	**88.36**

† denotes results we have reproduced. Models with * use difference features along with standard heatmaps and use separate models for all. P - pose heatmaps, PAF - Part affinity Fields [4], CF - Features from the pose extractor CNN.

Table 6 lists our results. We obtain improvements on all previously reported metrics over SIP-Net [55] while using only pose heatmaps and without relying on any tracking - further justifying our approach. Our approach also outperforms the pose-only model of [34] which uses a more accurate pose extractor and we obtain 2.7% improvement on top-5.

4.3. Additional Analyses

JMRN *vs*. Deeper Baseline. Our data augmentation strategy and model design helps us train deeper models and learn improved representations without over-fitting. To verify that our model leads to superior results compared to baseline due to design choices and not increased number of parameters, in Table 7 we experiment with a deeper version of the baseline model (PoTion). We add three convolutional layers to the baseline model, increasing the number of parameters in the baseline to match that of our model. For a fair comparison, we make use of our data augmentation scheme in the deeper baseline model too. The deeper baseline still performs worse than our proposed model, thus validating our proposed approach.

Table 5: Results on the validation set of the AVA v2.1 dataset. Results on this dataset demonstrate that the approach works on a large scale spatio-temporal action localization dataset. We see a significant improvement over the reproduced baseline. Further, we see that JMRN trained from scratch gives performance close to I3D-RGB model which was pretrained using large-scale Kinetics-400 dataset justifying the use of joint trajectories for action recognition. Fusion with a RGB-based model gives further gains.

Method	Pretraining	Modality	mAP
I3D [6]	Kinetics-400	RGB	14.5
I3D [6]	Kinetics-400	RGB + Flow	15.6
ACRN [44]	Kinetics-400	RGB + Flow	17.4
I3D [6]	Kinetics-600	RGB	21.9
SlowFast-R101-NL [15]	Kinetics-600	RGB	28.06†
PoTion[11]	None	Pose	13.1
Ours	None	Pose	14.1
Ours + SlowFast-R101-NL		Pose + RGB	**28.4**

† denotes results we have reproduced using the publicly available model.

Table 6: Mimetics experiments. Results when models are trained on the 50 Kinetics classes that intersect with Mimetics. We obtain an improvement over SIP-Net[55] while not requiring any tracking and using pose heatmaps alone.

Method	top 1	top 5	mAP
SIP-Net [55]	25.1	51.4	38.3
Ours	**26.6**	**52.7**	**40.0**

Table 7: JMRN *vs*. Deeper baseline. Our model outperforms a deeper version of the model proposed in [11] thus showing the effectiveness of our approach.

	JHMDB-1	HMDB-1	Charades
Deeper Baseline	66.38	49.25	14.88
Ours	**71.08**	**54.05**	**16.2**

Comparison with a Non-Siamese Network. Parameters of our motion extractor network are shared across all joints. Here we show the results when we do not share these parameters and instead learn separate motion extractors for

Table 8: Effect of sharing parameters of the motion extractor module. The non-Siamese model performs worse than our proposed model.

	JHMDB-1	HMDB-1	Charades
Non-Siamese	65.98	51.66	14.93
Ours	**71.08**	**54.05**	**16.2**

each joint. To maintain a similar number of parameters as the baseline, the motion extractors are made shallower than in JMRN. Table 8 shows the results of these experiments. This model performs worse than our proposed JMRN model and shows the importance of sharing the parameters.

What is the model learning? Since we first extract joint-motion signatures separately before fusing, we can attribute the weights in the selector to specific joints. In Fig. 2 we show the average per-class weights for the HMDB-1 test set. We see some interesting, distinctive patterns. There are some joints like `LHand` and `RHand` which are used by the model for most inputs. Since none of the actions requires an explicit focus on keypoints on the face, motion features for keypoints like eyes and ears tend to be suppressed by the model for most cases. We also see a couple of cases where the model focuses on a very few joints like sword exercise. This might be due to biases underlying the dataset wherein the person performing the action is often in a similar pose and relatively few joints are enough to recognize the action. Further, they might not add new information than that given by other joints. We also see some usage of limbs that make intuitive sense. For example, as expected, the activity 'eat' and 'drink' predominantly depends on the upper limbs. The design of JMRN model allows us to see which joints were predominantly used by the model to infer activities.

Alternative Pose Extractors Our approach is modular and allows use of any off-the-shelf pose extractor. This is unlike approaches like PA3D and SIP-Net which make use of features from pose extractor and require access to features from pose extractor. We also use a more accurate pose extractor AlphaPose [14, 25, 58] on the Charades dataset. JMRN with AlphaPose gives an improvement of +0.5 mAP over JMRN with OpenPose. This demonstrates our approach's modular nature and applicability to other pose extractors. For fair comparison with previous approaches we used OpenPose for the experiments in the paper.

Benefit of joint-reweighting Apart from the benefit of visualizing the joints that the model focuses on, the joint-selection helps improve performance too. We see that using the concatenated per-joint feature without reweighting drops the performance by 1% and 0.9 mAP on JHMDB and Charades respectively while having comparable performance on the HMDB dataset.

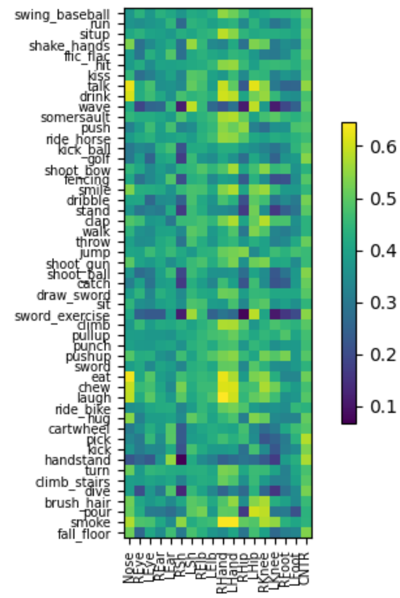

Figure 2: Visualization of input conditional weights w_i for HMDB-1 test set. We show the average weights for each class. Our approach of having a joint-selector as a part of the model allows for easy interpretation of joint importance for various actions. For example, we see that actions like 'eat' and 'drink' have a high dependence on hands.

5. Conclusion

We presented an approach to learn better representations for pose-based action recognition. The proposed JMRN extracts useful per-joint motion information before reweighing the information for collective inter-joint reasoning. Our novel joint-contrastive loss further improves the results. This leads to improved performance on the downstream task of action recognition. The proposed pose-aware data augmentation step, applies a cascade of global and group-wise jitter. Our proposed approaches lead to improvement over state-of-the-art on JHMDB, HMDB, Charades, AVA and Mimetics using pose heatmaps alone. Fusion with state-of-the-art RGB and flow-based model leads to further improvements showing complementary nature of the pose stream. Our proposed approach can be extended to explicitly handle missing joint information and people in the background. We leave these to future work.

6. Acknowledgements

The authors would like to thank Alex Hanson, Ketul Shah, Susmija Reddy and Anoop Cherian for their helpful suggestions. This research is partially supported by the ONR MURI Grant N00014-20-1-2787.

References

[1] Sadjad Asghari-Esfeden, Mario Sznaier, and Octavia Camps. Dynamic motion representation for human action recognition. In *The IEEE Winter Conference on Applications of Computer Vision*, 2020.

[2] Congqi Cao, Yifan Zhang, Chunjie Zhang, and Hanqing Lu. Action recognition with joints-pooled 3D deep convolutional descriptors. In *IJCAI*, 2016.

[3] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Openpose: realtime multi-person 2D pose estimation using part affinity fields. *arXiv preprint arXiv:1812.08008*, 2018.

[4] Zhe Cao, T. imon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2D pose estimation using part affinity fields. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[5] Joao Carreira, Pulkit Agrawal, Katerina Fragkiadaki, and Jitendra Malik. Human pose estimation with iterative error feedback. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4733–4742, 2016.

[6] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? A new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[7] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[8] Anoop Cherian, Basura Fernando, Mehrtash Harandi, and Stephen Gould. Generalized rank pooling for activity recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3222–3231, 2017.

[9] Anoop Cherian, Suvrit Sra, Stephen Gould, and Richard Hartley. Non-linear temporal subspace representations for activity recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2197–2206, 2018.

[10] Guilhem Chéron, Ivan Laptev, and Cordelia Schmid. P-CNN: Pose-based CNN features for action recognition. In *2015 IEEE International Conference on Computer Vision*, 2015.

[11] Vasileios Choutas, Philippe Weinzaepfel, Jérôme Revaud, and Cordelia Schmid. Potion: Pose motion representation for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[12] Wenbin Du, Yali Wang, and Yu Qiao. RPAN: an end-to-end recurrent pose-attention network for action recognition in videos. In *IEEE International Conference on Computer Vision, ICCV*, 2017.

[13] Wenbin Du, Yali Wang, and Yu Qiao. Rpan: An end-to-end recurrent pose-attention network for action recognition in videos. *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017.

[14] Hao-Shu Fang, Shuqin Xie, Yu-Wing Tai, and Cewu Lu. Rmpe: Regional multi-person pose estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, 2017.

[15] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6202–6211, 2019.

[16] Georgia Gkioxari and Jitendra Malik. Finding action tubes. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[17] Chunhui Gu, Chen Sun, David A Ross, Carl Vondrick, Caroline Pantofaru, Yeqing Li, Sudheendra Vijayanarasimhan, George Toderici, Susanna Ricco, Rahul Sukthankar, et al. AVA: A video dataset of spatio-temporally localized atomic visual actions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6047–6056, 2018.

[18] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018.

[19] H. Jhuang, J. Gall, S. Zuffi, C. Schmid, and M. J. Black. Towards understanding action recognition. In *International Conf. on Computer Vision (ICCV)*, 2013.

[20] Gunnar Johansson. Visual perception of biological motion and a model for its analysis. *Perception & psychophysics*, 1973.

[21] M. E. Kalfaoglu, Sinan Kalkan, and Aydin Alatan. Late temporal modeling in 3D CNN architectures with bert for action recognition. *ArXiv*, abs/2008.01232, 2020.

[22] Lipeng Ke, Ming-Ching Chang, Honggang Qi, and Siwei Lyu. Multi-scale structure-aware network for human pose estimation. In *Proceedings of the european conference on computer vision (ECCV)*, pages 713–728, 2018.

[23] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 18661–18673. Curran Associates, Inc., 2020.

[24] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. Hmdb: A large video database for human motion recognition. In *2011 International Conference on Computer Vision*, 2011.

[25] Jiefeng Li, Can Wang, Hao Zhu, Yihuan Mao, Hao-Shu Fang, and Cewu Lu. Crowdpose: Efficient crowded scenes pose estimation and a new benchmark. *arXiv preprint arXiv:1812.00324*, 2018.

[26] Yingwei Li, Yi Li, and Nuno Vasconcelos. Resound: Towards action recognition without representation bias. In *ECCV*, 2018.

[27] Yi Li and Nuno Vasconcelos. Repair: Removing representation bias by dataset resampling. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[28] Mengyuan Liu and Junsong Yuan. Recognizing human actions as the evolution of pose estimation maps. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[29] Dennis Ludl, Thomas Gulde, and Cristóbal Curio. Simple yet efficient real-time pose-based action recognition. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, 2019.

[30] Yue Luo, Jimmy S. J. Ren, Zhouxia Wang, Wenxiu Sun, Jinshan Pan, Jianbo Liu, Jiahao Pang, and Liang Lin. LSTM pose machines. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018.

[31] Diogo C Luvizon, David Picard, and Hedi Tabia. 2D/3D pose estimation and action recognition using multitask deep learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[32] William J. McNally, Alexander Wong, and John McPhee. Star-net: Action recognition using spatio-temporal activation reprojection. In *16th Conference on Computer and Robot Vision, CRV 2019, Kingston*, 2019.

[33] Gyeongsik Moon, Ju Yong Chang, and Kyoung Mu Lee. Posefix: Model-agnostic general human pose refinement network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7773–7781, 2019.

[34] Gyeongsik Moon, Heeseung Kwon, Kyoung Mu Lee, and Minsu Cho. Integralaction: Pose-driven feature integration for robust human action recognition in videos. *ArXiv*, abs/2007.06317, 2020.

[35] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[36] Xiaojiang Peng and Cordelia Schmid. Multi-region two-stream R-CNN for action detection. In *ECCV*, 2016.

[37] Tomas Pfister, James Charles, and Andrew Zisserman. Flowing convnets for human pose estimation in videos. *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015.

[38] AJ Piergiovanni, Anelia Angelova, Alexander Toshev, and Michael S Ryoo. Evolving space-time neural architectures for videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1793–1802, 2019.

[39] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Two-stream adaptive graph convolutional networks for skeleton-based action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[40] Gunnar A Sigurdsson, Santosh Divvala, Ali Farhadi, and Abhinav Gupta. Asynchronous temporal fields for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[41] Gunnar A. Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. 2016.

[42] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in neural information processing systems*, 2014.

[43] Sijie Song, Cuiling Lan, Junliang Xing, Wenjun Zeng, and Jiaying Liu. An end-to-end spatio-temporal attention model for human action recognition from skeleton data. In *Thirty-first AAAI conference on artificial intelligence*, 2017.

[44] Chen Sun, Abhinav Shrivastava, Carl Vondrick, Kevin Murphy, Rahul Sukthankar, and Cordelia Schmid. Actor-centric relation network. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 318–334, 2018.

[45] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5693–5703, 2019.

[46] Jonathan Tompson, Arjun Jain, Yann LeCun, and Christoph Bregler. Joint training of a convolutional network and a graphical model for human pose estimation. In *Advances in Neural Information Processing Systems*, 2014.

[47] Alexander Toshev and Christian Szegedy. Deeppose: Human pose estimation via deep neural networks. *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[48] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2018.

[49] Raviteja Vemulapalli, Felipe Arrate, and Rama Chellappa. Human action recognition by representing 3D skeletons as points in a lie group. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014.

[50] Jue Wang, Anoop Cherian, Fatih Porikli, and Stephen Gould. Video representation learning using discriminative pooling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[51] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *European conference on computer vision*, 2016.

[52] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018.

[53] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Convolutional pose machines. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 4724–4732, 2016.

[54] Yi Wei, Wenbo Li, Yanbo Fan, Linghan Xu, Ming-Ching Chang, and Siwei Lyu. 3D single-person concurrent activity detection using stacked relation network. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence*, 2020.

[55] Philippe Weinzaepfel and Grégory Rogez. Mimetics: Towards understanding human actions out of context. *International Journal of Computer Vision*, 129(5):1675–1690, 2021.

[56] Chao-Yuan Wu, Christoph Feichtenhofer, Haoqi Fan, Kaiming He, Philipp Krahenbuhl, and Ross Girshick. Long-term feature banks for detailed video understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[57] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[58] Yuliang Xiu, Jiefeng Li, Haoyu Wang, Yinghong Fang, and Cewu Lu. Pose Flow: Efficient online pose tracking. In *BMVC*, 2018.

[59] An Yan, Yali Wang, Zhifeng Li, and Yu Qiao. PA3D: Pose-action 3D machine for video recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[60] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Thirty-second AAAI conference on artificial intelligence*, 2018.

[61] Wei Yang, Shuang Li, Wanli Ouyang, Hongsheng Li, and Xiaogang Wang. Learning feature pyramids for human pose estimation. In *proceedings of the IEEE international conference on computer vision*, pages 1281–1290, 2017.

[62] Kaiyu Yue, Ming Sun, Yuchen Yuan, Feng Zhou, Errui Ding, and Fuxin Xu. Compact generalized non-local network. In *Advances in Neural Information Processing Systems*.

[63] Mohammadreza Zolfaghari, Gabriel L Oliveira, Nima Sedaghat, and Thomas Brox. Chained multi-stream networks exploiting pose, motion, and appearance for action classification and detection. In *Proceedings of the IEEE International Conference on Computer Vision*, 2017.