

Weakly-Supervised Multi-Person Action Recognition in 360° Videos

Junnan Li^{*1}, Jianquan Liu², Yongkang Wang¹, Shoji Nishimura², and Mohan S. Kankanhalli¹

¹School of Computing, National University of Singapore

²NEC Corporation

lijunnan@u.nus.edu, j-liu@ct.jp.nec.com, yongkang.wong@nus.edu.sg,
s-nishimura@bk.jp.nec.com, mohan@comp.nus.edu.sg

Abstract

The recent development of commodity 360° cameras have enabled a single video to capture an entire scene, which endows promising potentials in surveillance scenarios. However, research in omnidirectional video analysis has lagged behind the hardware advances. In this work, we address the important problem of action recognition in top-view 360° videos. Due to the wide field-of-view, 360° videos usually capture multiple people performing actions at the same time. Furthermore, the appearance of people are deformed. The proposed framework first transforms top-view omnidirectional videos into panoramic videos using a calibration free method. Then spatial-temporal features are extracted using region-based 3D CNNs for action recognition. We propose a weakly-supervised method based on multi-instance multi-label learning, which trains the model to recognize and localize multiple actions in a video using only video-level action labels as supervision. We perform experiments to quantitatively validate the efficacy of the proposed method over state-of-the-art baselines and variants of our model, and qualitatively demonstrate action localization results. To enable research in this direction, we introduce the 360Action dataset. It is the first omnidirectional video dataset for multi-person action recognition with a diverse set of scenes, actors and actions. The dataset is available at <https://github.com/ryukenzen/360action>.

1. Introduction

Omnidirectional cameras can monitor a vast scene with a small budget. Recently, commodity omnidirectional cameras such as Samsung Gear 360 and Kodak PixPro SP360 have been developed, which can capture high-quality 4K videos. A single top-view omnidirectional camera covers the same area as multiple conventional cameras, making it a preferable device in surveillance scenarios. Besides being cost-efficient and easier to install, an omnidirectional cam-

era requires only one algorithm to analyze the entire scene, which avoids the inconvenience of synchronization and coordination among multiple conventional cameras, and reduces security risks of privacy attack.

Despite the huge potential of omnidirectional cameras for video surveillance, 360° video analysis has received limited attention. In this paper, we address the important problem of action recognition in 360° videos. There are two challenges arising from 360° videos that make state-of-the-art deep network based action recognition algorithms ineffective. First, the appearance of people are deformed. Specifically, people would be rotated at varying angles, thus making a deep network pretrained on standard perspective videos unable to extract useful features. In this paper, we propose a method to transform an omnidirectional video into a panoramic video where people stand upright. Our method is calibration-free, easy to implement, and does not require any training.

The wide field-of-view (FoV) of omnidirectional cameras results in the second challenge for action recognition. In a practical scenario where the camera is installed at a place with large pedestrian volume, the videos are likely to capture many people performing actions at the same time. Since it is computational intensive to analyze each person individually, an efficient method should be able to simultaneously recognize actions for multiple people. Furthermore, from the perspective of curating training data, it is both expensive and time-consuming to extensively annotate each person's position (*i.e.* bounding box) and action. On the other hand, it is much easier to acquire annotation only for the video-level action labels without linking each action to a specific person.

In this work, we propose a weakly-supervised method for multi-person action recognition in high-resolution videos. Our model is weakly-supervised in the sense that it is trained using only video-level action labels. We formulate the problem as multi-instance multi-label (MIML) learning, and exploit two intuitions to facilitate the design of our framework: (1) *only a fraction of regions in the video are informative for a certain action*, and (2) *one person can*

^{*}This work is mainly completed during Junnan Li's period of internship at NEC Corporation

only perform one action at a time.

Our key contributions can be summarized as follows:

- We propose a novel framework for multi-person action recognition in top-view omnidirectional videos. The first step of our framework addresses people’s rotational deformation by using a calibration-free method to transform omnidirectional videos into panoramic videos.
- The second step of our framework achieves multi-person action recognition with only video-level labels as supervision. We propose region-based 3D CNN to extract informative spatial-temporal features, and divide the spatial-temporal features into multiple instances for multi-label learning. Our weakly-supervised model can learn to not only recognize but also localize each action in the video.
- We introduce 360Action, the first omnidirectional video dataset for action recognition. Our dataset contains high-resolution videos recorded in diverse scenes. 360Action paves the way for future research in omnidirectional video analysis.
- We perform multi-person action recognition experiments on 360Action dataset. We quantitatively validate the efficacy of the proposed method, and qualitatively demonstrate action localization results. Furthermore, we conduct ablation study to analyze several model design choices.

2. Related Work

2.1. Action Recognition

Action recognition is a long-standing problem in computer vision and has been extensively studied. Recent approaches based on deep networks trained using large-scale video datasets have achieved great progress [20]. One line of approaches use two-stream Convolutional Neural Networks (CNNs) to process RGB and optical flow frames as appearance and motion information, respectively [27, 31, 8, 10, 9]. Another line of approaches focus on CNNs with 3D convolutional kernels [15, 30, 4]. 3D CNNs can effectively extract spatial-temporal features directly from videos. Very recently, Hara *et al.* [11, 12] trained deep 3D CNNs using the large-scale Kinetics dataset [16], and achieve state-of-the-art action recognition performance on multiple benchmarks. In this work, we use 3D CNNs as our backbone architecture for spatial-temporal feature extraction.

2.2. Omnidirectional Video Analysis

Due to the effectiveness of omnidirectional cameras for video surveillance, researcher have studied omnidirectional pedestrian detection, where the main challenge is pedestrian’s rotational deformation. Some works [25, 17] propose to first transform omnidirectional images into perspective images and then apply standard pedestrian detec-

tors. However, the transformation relies heavily on calibrated camera parameters, which requires user-interaction. Other works [29, 22] train orientation-aware networks using pedestrian images with synthetic rotation. However, such training introduces large computational cost and the trained network is biased towards the data used. In this paper, we propose a method to transform omnidirectional videos into panoramic videos where people stand upright. Our method does not require camera calibration nor training.

Unlike conventional perspective videos, only a limited amount of work has studied action recognition in omnidirectional videos [5, 28, 1]. Previous methods use constrained data and have limited performance. Our work is the first to exploit the representation power of deep networks for multi-person action recognition in 360° videos. We also introduce a new benchmark dataset to enable future research in this direction.

Besides the video domain, 360° data has also been exploited to learn visual representations in a self-supervised manner [18], which achieves improved performance on many vision tasks.

2.3. Multiple Instance Learning

Multiple instance learning (MIL) is first introduced by Dietterich *et al.* [7] for drug activity prediction. In MIL framework, each example consists of a bag of instances, and only the bag-level label is available. Recently, researchers have incorporated deep networks into the MIL framework for weakly-supervised image classification [32], object detection [6, 14], co-saliency detection [34], social relationship recognition [19], and image segmentation [23].

Multi-instance multi-label learning is a variant of MIL where an example with multiple instances is also associated with multiple labels [36, 37]. In our problem, we have multiple regions as instances from a video, and only the video-level action labels are available as weak supervision.

3. Omnidirectional to Panoramic

To address the rotational deformation of people in omnidirectional videos, we introduce a method to transform a top-view omnidirectional video into a panoramic one, so that CNNs can be used for feature extraction. Our method is *calibration free*. It does not require access to camera parameters or the configuration of the camera, therefore is more applicable in practical scenarios. Next we delineate the details.

First, the dimensions of the panorama have a proportional relationship with the FoVs of the camera:

$$\frac{h}{w} = \frac{\text{VFoV}}{2 \times \text{HFoV}}, \quad (1)$$

where w and h are the width and height of the panorama, respectively. HFoV is the horizontal field-of-view (*i.e.*, 360°)

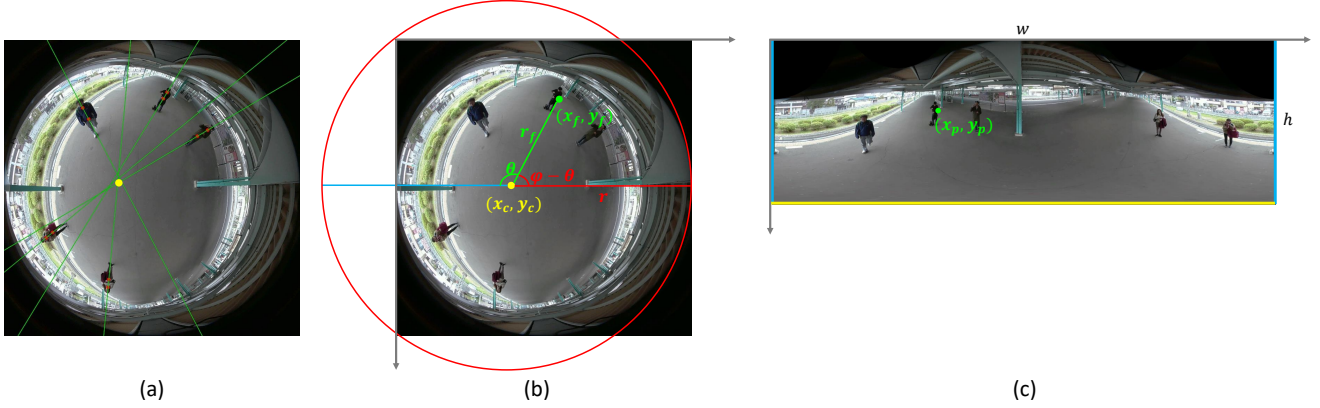


Figure 1: (a) We locate the center (yellow) as the point closest to all spine lines (lines that are approximately perpendicular to the ground). (b&c) We establish a mapping from any pixel (x_p, y_p) in the panoramic frame to a pixel (x_f, y_f) in the fisheye frame.

of the fisheye camera and VFoV is the vertical field-of-view (*i.e.*, 235°). In this work we set $h = 800$, which leads to $w = 2451$ according to the above equation.

Our method exploits this observation: in an omnidirectional frame captured by a top-view fisheye camera, straight lines that are perpendicular to the ground would all intersect at a single point, which we refer to as the center c . In order to locate its pixel coordinate (x_c, y_c) , we detect the spines of people standing upright and use the spines to approximately represent lines perpendicular to the ground. As shown in Figure 1a, we use Mask-RCNN [13] to detect two keypoints for each person: mid-shoulder and mid-hip. Then we connect the two keypoints to acquire the spine for a person, represented by the line $k_i x + y + z_i = 0$. Note that we rotate the frame by $\{0^\circ, 90^\circ, 180^\circ, 270^\circ\}$ and repeat the spine detection, so that people at different orientations can be utilized.

Having detected K spine lines, we find c as the point with the smallest total distance *w.r.t* all lines:

$$(x_c, y_c) = \arg \min_{(x, y)} \sum_i^K \frac{|k_i x + y + z_i|}{\sqrt{k_i^2 + 1}} \quad (2)$$

Note that we can compute (x_c, y_c) for multiple frames and average the coordinates for a more accurate center location.

Next, we need to find a mapping from a pixel $p = (x_p, y_p)$ in the panorama to its corresponding pixel $f = (x_f, y_f)$ in the fisheye frame. In the fisheye frame (see Figure 1b), we denote the distance between c and its furthest frame boundary as r , and the distance between f and c as r_f . A user-defined angle (ϕ) determines the blue starting line from which we unwrap the fisheye frame. θ denotes the angle between the blue line and the green line. We can establish a mapping from (x_p, y_p) to (θ, r_f) as

$$\frac{x_p}{w} = \frac{\theta}{360}, \quad \frac{h - y_p}{h} = \frac{r_f}{r} \quad (3)$$

Then we can map the polar coordinate (θ, r_f) to the Cartesian coordinate (x_f, y_f) with

$$\frac{x_f - x_c}{r_f} = \cos(\phi - \theta), \quad \frac{y_c - y_f}{r_f} = \sin(\phi - \theta) \quad (4)$$

Thus, we have successfully established the mapping $(x_p, y_p) \rightarrow (\theta, r_f) \rightarrow (x_f, y_f)$. Note that if the corresponding f for a p is outside the fisheye frame, we assign black color to p .

4. Multi-person Action Recognition

Given a high-resolution panoramic video containing multiple people, we propose a method to recognize all actions in the video with one forward pass. Our method is weakly-supervised, where only video-level action labels are available during training. An overview of the proposed framework is shown in Figure 2. It consists of two steps: spatial-temporal feature extraction and multi-instance multi-label learning. Next we explain our method in details.

4.1. Region-based 3D CNN

Deep 3D CNNs trained on large-scale video datasets can learn representative spatial-temporal features [12]. Therefore, we utilize 3D ResNet-34 [11] pretrained on Kinetics [16] to extract features for each 16-frame clip. Previous methods on action recognition normally resize the videos to heights of 240 pixels. However, since our videos have a much wider FoV and higher resolution than conventional videos, such resizing would make the people extremely small and the actions unrecognizable. Therefore, we use the original video with size 800×2451 as input. The resulting convolutional feature map from the conv5_x layer has a spatial size of 25×77 .

A significant portion of the video contains only background and does not provide useful information for the ac-

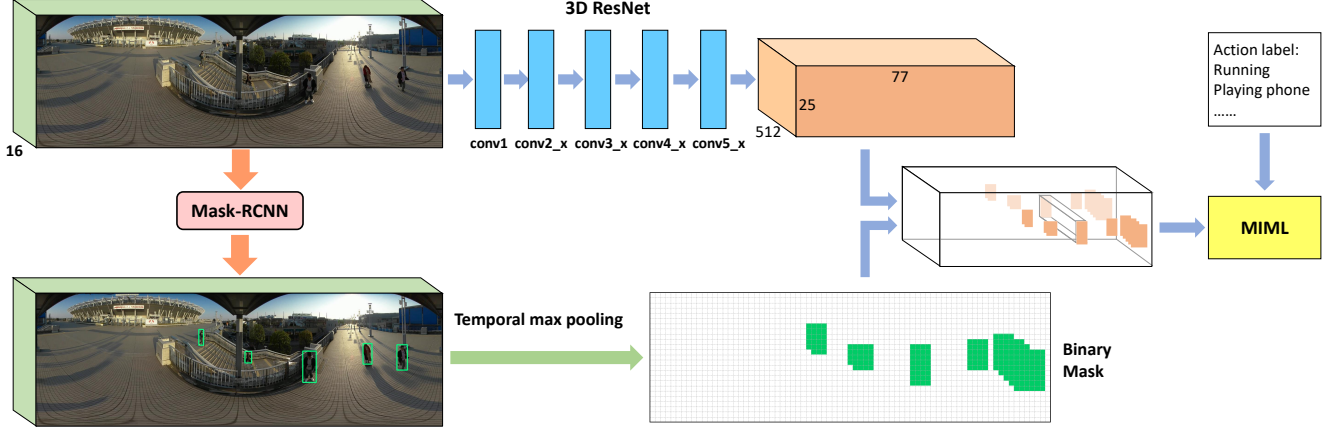


Figure 2: Overview of the proposed action recognition framework. We use 3D ResNet to extract a convolutional feature map for a input video clip. Uninformative features are set as zeros by applying a binary mask calculated with person detection. The masked convolutional feature map is used for multi-instance multi-label learning.

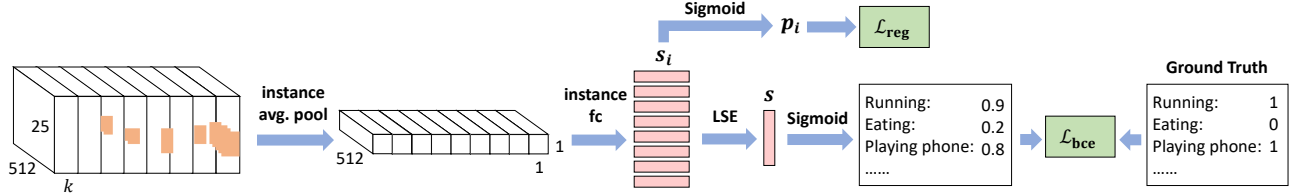


Figure 3: The proposed Multi-instance Multi-label Learning (MIML) method. The masked convolutional feature map is divided into multiple instances. Each instance outputs a set of scores s_i for the possible actions. All instance-level scores are then aggregated with the Log-Sum-Exp (LSE) function. We add a regularization term (\mathcal{L}_{reg}) to the cross entropy loss (\mathcal{L}_{bce}), which penalizes the model if a single instance outputs high scores for multiple actions.

tions. We would like to discard those uninformative and potentially misleading features. To this end, we apply a person detector (*i.e.*, Mask-RCNN [13]) to acquire the bounding boxes for all people in each frame (examples are provided in the supplementary material). Then we max-pool the bounding boxes across all 16 frames in a clip to get a binary mask. This mask is robust to false negative detection in certain frames because the person can be detected in other frames within the same clip. We resize the mask to the same width and height as the convolutional feature map, and multiply the feature map with the mask so that only features in the masked region are preserved while others are set as zero. The masked spatial-temporal features are used for learning actions.

4.2. Multi-instance Multi-label Learning

As shown in Figure 3, given a masked convolutional feature map, we split it into N blocks where each block has width k (features are zero-padded). We refer to each block as an instance i , and apply spatial average pooling on each instance to acquire a feature of size $512 \times 1 \times 1$.

Each instance-level feature is then flattened and processed by a fully-connected (fc) layer to generate a set of scores $s_i = \{s_i^a\}, \forall a \in \mathcal{C}$ for the set of action classes \mathcal{C} .

For a ground-truth action a , an obvious way to aggregate its instance-level scores is to take the average across all instances:

$$s^a = \frac{1}{N} \sum_i s_i^a \quad (5)$$

However, this would assign the same weights to all instances, even the ones that are not relevant to the action. In fact, only one or a few instances are responsible for the occurrence of a . Another aggregation function is the max function:

$$s^a = \max_i s_i^a \quad (6)$$

This would consider only one instance to be responsible for the occurrence of the action, neglecting the case where multiple people from different instances can perform the same action a .

To address this limitation, we use a smooth version of

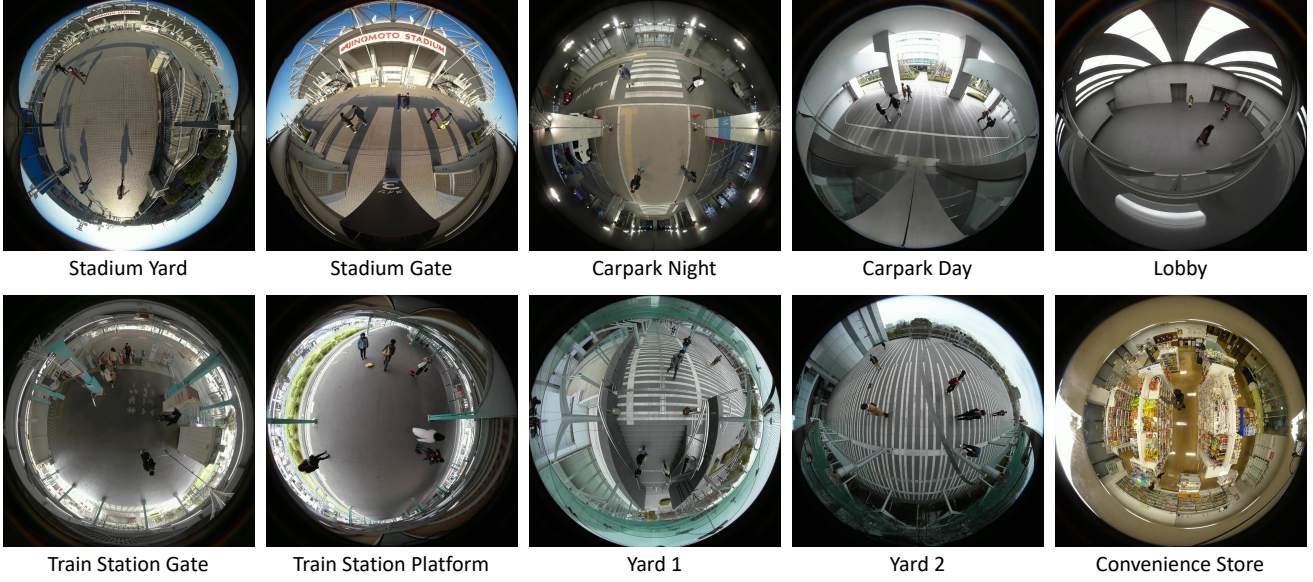


Figure 4: Sample frames of the 10 diverse scenes in 360Action dataset. Each video captures multiple people performing actions.

the max function, called Log-Sum-Exp (LSE) [3]:

$$s^a = \frac{1}{r} \log \left[\frac{1}{N} \sum_i \exp(rs_i^a) \right] \quad (7)$$

We can use r to control how smooth the function is. Larger r would have a similar affect to max and consider the most important instance, whereas smaller r would consider other less important instances. In our experiment we set $r = 0.8$ (see Section 6.2 for ablation study on hyper-parameters).

The aggregated scores are then scaled to $[0, 1]$ with the sigmoid function:

$$p^a = \frac{1}{1 + e^{-s^a}} \quad (8)$$

During training, the label y_a for an action a takes the value 1 if the action exists in the video and 0 otherwise. We train our model to minimize the binary cross-entropy loss:

$$\mathcal{L}_{\text{bce}} = - \sum_a (y^a \log(p^a) + (1 - y^a) \log(1 - p^a)) \quad (9)$$

Sparsity Regularization. Since one person can only perform one action at a time, we propose a regularization term which penalizes the model if a single instance outputs high scores for multiple actions. To this end, we first apply sigmoid function to scale each instance’s score s_i^a to $[0, 1]$:

$$p_i^a = \frac{1}{1 + e^{-s_i^a}} \quad (10)$$

Then, the regularization term is defined as:

$$\mathcal{L}_{\text{reg}} = \sum_i \frac{\sum_a p_i^a - \max_a p_i^a}{\max_a p_i^a} \quad (11)$$

Minimizing \mathcal{L}_{reg} would encourage each instance to output a high score only for the action that it is most confident about.

During training, The total loss to minimize is:

$$\mathcal{L} = \mathcal{L}_{\text{bce}} + \alpha \mathcal{L}_{\text{reg}}, \quad (12)$$

where α controls the strength of regularization and is set as 0.001 in our experiment (value determined by validation).

4.3. Action Localization

After training the 3D CNN following the proposed method, we can localize the predicted actions by finding the areas in the convolutional feature map that are relevant to certain predictions. We exploit Grad-CAM [26] for weakly-supervised action localization, which work as follows.

Given a test video clip, the trained model outputs a set of action scores $\{p^a\}, \forall a \in \mathcal{C}$. For each predicted action a with $p^a > 0.5$, we apply back-propagation to calculate the gradient of p^a with respect to the k -th feature map A^k of the last convolutional layer (conv5_x), i.e., $\frac{\partial p^a}{\partial A^k}$. The gradients are global-average-pooled to obtain a weight α_k^a , which captures the importance of A^k for action a :

$$\alpha_k^a = \frac{1}{Z} \sum_i \sum_j \frac{\partial p^a}{\partial A_{ij}^k} \quad (13)$$

The convolutional feature maps are then weighted combined to obtain a heatmap of the same size as A^k (i.e., 25×77 for 3D ResNet-34):

$$H = \text{ReLU} \left(\sum_k \alpha_k^a A^k \right) \quad (14)$$

High values at certain positions in the heatmap indicate positive influence of that area to the action of interest. We resize H to the same size as the panoramic video frames (*i.e.*, 800×2451) to visualize the action localization results. Examples are shown in Figure 7.

5. 360Action Dataset

In this section, we introduce 360Action, the first omnidirectional video dataset for action recognition. Different from conventional action recognition datasets, 360Action contains 360° videos which capture multiple person performing multiple actions at the same time. The dataset contains 784 diverse videos, recorded using a state-of-the-art omnidirectional camera, Kodak PixPro SP360 4K, which has a horizontal FoV of 360° and a vertical FoV of 235° . All videos have a high resolution of 2880×2880 . Next we delineate the details of our dataset.

Scenes. 360Action consists of videos from 10 diverse scenes including 2 indoors scenes (lobby, convenience store) and 8 outdoor scenes (stadium yard, stadium gate, carpark night, carpark day, train station gate, train station platform, yard 1 and yard 2). Example frames from each scene are shown in Figure 4. Different scenes have different environments (e.g. train station platform has moving trains in the background whereas carpark has cars in the background) and different lighting conditions (e.g. daytime sunny, daytime cloudy, indoor lighting, etc.). The diversity of scenes puts high requirement on the algorithm to be robust to different conditions.

Subjects. We hired 80 different subjects for video recording. 40 subjects are male and 40 subjects are female. The ages of the subjects range from 10 to 40 years old. Each subject is assigned a consistent ID number across all videos. We split subjects with IDs 1-60 into training set, and subjects with IDs 61-80 into test set.

Actions. The dataset contains 19 classes of daily actions, including 15 single-person actions (eating, wearing jacket, walking, waving, etc.) and 4 interactions (pushing, handshaking, taking something, giving something). During recording, we assign each subject a scripted set of actions to perform. Each subject performs each action at least once in a scene. In each video, multiple subjects perform multiple actions concurrently. The maximum number of concurrent actions in a video is 7. On average, each video contains 4 concurrent actions.

6. Experiment

6.1. Implementation Details

Our model is trained using stochastic gradient descent (SGD) with a momentum of 0.9. We use a batch size of 32 clips and a learning rate of 0.01. The learning rate is decayed by half every 10 epochs, and the model is trained

Table 1: Comparison with state-of-the-art methods for weakly-supervised multi-person action recognition on 360Action dataset.

Method	mAP (%)
Collective [2]	61.27
3D ResNet [11]	61.95
R-C3D [33]	58.74
MiCT [35]	62.18
Ours	70.12

for 50 epochs in total. All hyper-parameters are determined via cross-validation.

6.2. Action Recognition

Comparison with state-of-the-art methods. First, we compare the proposed model with multiple state-of-the-art action recognition methods [2, 33, 11, 35]. We modify these methods for our task of weakly-supervised multi-person action recognition. Specifically, we apply multi-label classification loss for all baselines. We also remove the detection loss in [2] because we do not assume to have detection ground-truth. The comparison results are shown in Table 1. We report the mean average precision (mAP) across all classes, which is the standard evaluation metric that takes both precision and recall into account. Our MIML-based method significantly outperforms the baselines due to its unique ability to discover associations between instances and actions.

Ablation study. Next, we conduct ablation study by comparing the proposed model with its multiple variants. For each baseline, we remove one component from the proposed model while keeping others components unchanged, so that we can examine the effect of each proposed component for weakly-supervised multi-person action recognition. The variants are delineated as follows.

- Without mask: we do not use the binary region mask (see Section 4.1) to filter out useless information. In this case the MIML module receives a noisy input.
- Avg-pool: we do not use the proposed MIML module (see Section 4.2) to acquire the action scores. Instead, we apply average pooling on the masked feature map from conv5.x layer, and use a fully-connected layer to transform the avg-pooled feature into action scores.
- Max-pool: we do not use the proposed MIML module to acquire the action scores. Instead, we apply max pooling on the masked feature map from conv5.x layer, and use a fully-connected layer to transform the max-pooled feature into action scores.
- MIML-avg: we use the average function (see Eqn. 5) rather than LSE to aggregate instance-level action scores.

Table 2: Per-class average precision of the 19 actions.

Method	Walk	Run	Wave Hand	Pickup sth.	Drop sth.	Take off jacket	Wear jacket	Phone call	Play phone	Eat snack	Walk upstairs	Walk downstairs	Drink water	Tap out station	Tap in station	Handshake	Take sth.	Give sth.	Push
Max-pool	87.6	77.2	82.5	55.6	40.6	59.0	53.8	45.3	50.7	49.7	74.2	59.9	48.4	98.1	88.0	62.8	85.4	56.0	69.6
Ours	89.8	93.8	81.2	65.4	51.4	69.4	67.0	44.9	48.0	39.6	77.9	79.6	48.5	97.5	95.4	63.7	86.3	56.7	73.5

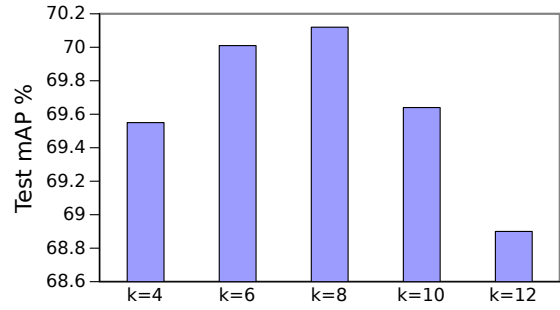
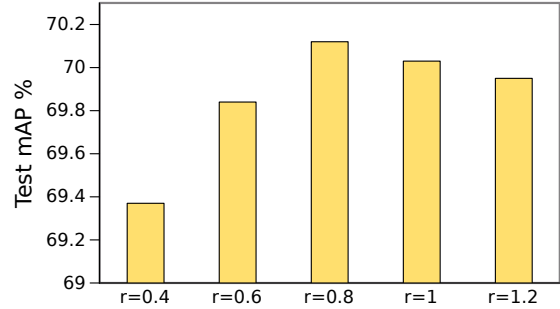
Table 3: Comparison with multiple variants of the proposed method on 360Action dataset. The results validate the efficacy of each proposed component (*i.e.*, region mask, MIML, sparsity regularization).

Method	mAP (%)
Without mask	67.52
Avg-pool	64.73
Max-pool	65.50
MIML-avg	68.59
MIML-max	69.07
MIML-attention	69.46
Without \mathcal{L}_{reg}	69.25
Ours (MIML-LSE)	70.12

- MIML-max: we use the max function (see Eqn. 6) rather than LSE to aggregate instance-level action scores.
- MIML-attention: we use the attention pooling method [21] to aggregate instance-level action scores. For each instance, a scalar attention weight is generated by feeding its feature to a fully-connected layer. The attention weights across all instances are then normalized using a Softmax function.
- Without \mathcal{L}_{reg} : we remove the regularization term that forces sparsity in instance-level scores (see Eqn. 12).

Table 3 shows the comparison results. The proposed MIML framework achieves a significant improvement of +4.52% over the standard feature max-pooling method. Comparing the four instance-level score aggregation methods, LSE performs better than attention pooling, max and average. Using region mask improves the performance by +2.5%. The proposed sparsity regularization can further improve the performance by +0.87%.

In Table 2, we show the per-class average precision (AP) for our model and the max-pool baseline. Generally speaking, actions that involves subtle movements (*e.g.*, play phone, eat snack) have lower APs, whereas actions with larger motions (*e.g.*, run, wave hand) are more easily recognized by the model.

Figure 5: The effect of k (*i.e.*, the width of each instance in the conv5_x feature map) on action recognition performance.Figure 6: The effect of r (which controls how smooth the LSE function is) on action recognition performance.

Hyper-parameters. Here we examine the effects of two important hyper-parameters in our MIML framework: k , the width of each instance in the conv5_x feature map; and r , the parameter that controls how smooth the LSE aggregation function is (see Eqn. 7).

Figure 5 shows the result of our method using different values of k . Larger k would result in fewer instances and more actions per instance, whereas smaller k leads to more instances with many instances containing no action. $k = 8$ achieves the best performance in our experiment.

Figure 6 shows the result of our method using different values of r . Larger r would make the LSE function more similar to max whereas smaller r would make it more similar to average. In our experiment $r = 0.8$ achieves the best

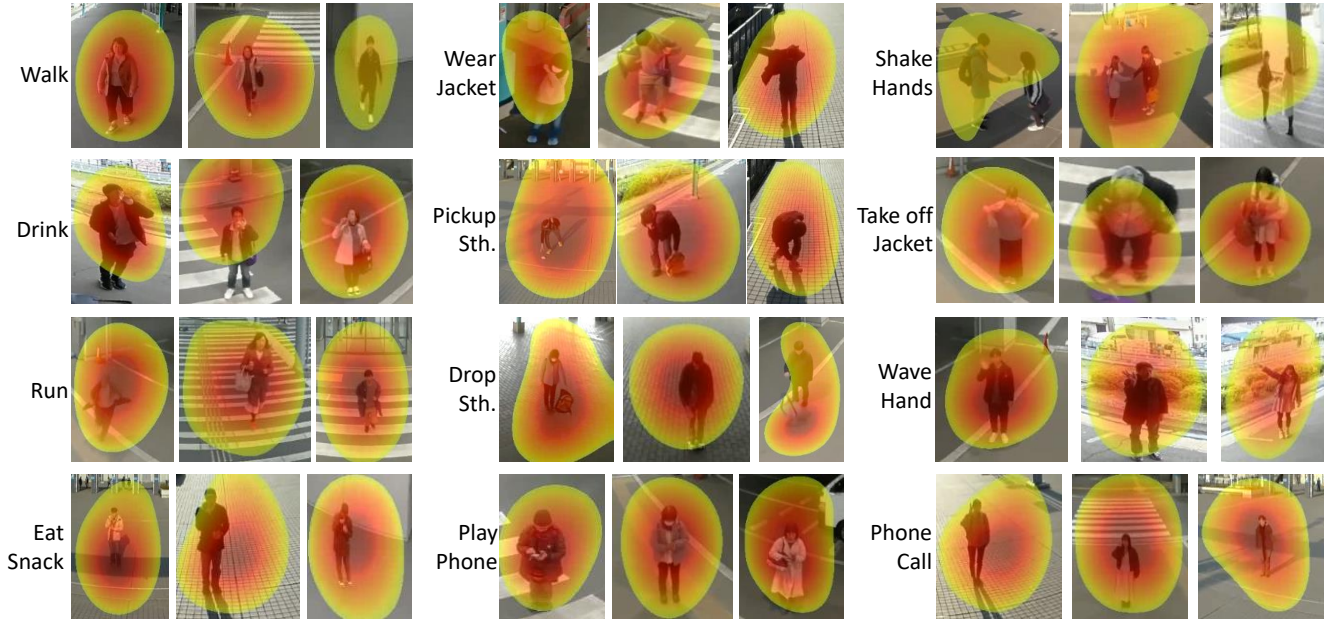


Figure 7: Qualitative results for weakly-supervised action localization. We employ GradCAM [26] to generate heatmaps that highlight the important areas for predicting certain actions. Our model trained with only video-level labels learns to not only predict the multiple actions but also coarsely localize each of its predicted actions.

Table 4: Inference speed analysis using different models for spatial-temporal feature extraction and person detection.

Feature Extractor	Person Detector	FPS	mAP (%)
3D ResNet-18	YOLO	6.28	66.17
3D ResNet-34	Mask-RCNN	1.14	70.12

performance.

Speed Analysis. During training, we do not finetune the 3D CNN and only perform feature extraction once for the entire training set. Therefore, the training process can be completed within 2 hours using a single NVIDIA V100 GPU. However, the inference speed is limited by the high resolution of the input videos. We perform an inference speed analysis in Table 4. By using a shallower feature extractor (*i.e.* 3D ResNet-18) and a single-stage person detector (*i.e.* YOLO [24]), we can achieve a much faster inference speed of 6.28 FPS at the cost of lower recognition performance.

6.3. Action Localization

We follow the method in Section 4.3 to calculate heatmaps for our model’s predicted actions. In Figure 7, we show qualitative results of the heatmaps overlaid on cropped video frames. Despite being trained using only action labels without any location information about where each action takes place, the proposed model can learn to associate its

predicted actions to specific people in the input videos, thus achieving weakly-supervised action localization.

7. Conclusion

To conclude, this paper aims to fill the gap between the rapid hardware development of 360° cameras and the limited progress in omnidirectional video analysis. We recognize the significant potential of 360° cameras in surveillance scenarios, and address the important task of action recognition in top-view omnidirectional videos. In face of the new challenges brought by the deformed and high-resolution videos, we propose a framework that achieves multi-person action recognition in one forward pass. To mitigate the difficulty of acquiring dense annotations, our method learns to recognize and localize actions in a weakly supervised manner where only video-level labels are required in training. In addition, to facilitate research in this direction, we curated 360Action, the first 360° video dataset with a diverse set of scenes and actors. We hope to see future research in omnidirectional videos to achieve the same level of success as in conventional videos.

Acknowledgment

This publication has emanated from the research at NEC Corporation, and is also partially supported by the National Research Foundation, Prime Minister’s Office, Singapore under its Strategic Capability Research Centres Funding Initiative.

References

- [1] T. Ang, A. W. Tan, C. Loo, and W. Wong. Wavelet mach filter for omnidirectional human activity recognition. *IJICIC*, 8:1–20, 2012. 2
- [2] T. M. Bagautdinov, A. Alahi, F. Fleuret, P. Fua, and S. Savarese. Social scene understanding: End-to-end multi-person action localization and collective activity recognition. In *CVPR*, pages 3425–3434, 2017. 6
- [3] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge university press, 2004. 5
- [4] J. Carreira and A. Zisserman. Quo vadis, action recognition? A new model and the kinetics dataset. In *CVPR*, pages 4724–4733, 2017. 2
- [5] X. Chen and J. Yang. Towards monitoring human activities using an omnidirectional camera. In *IEEE International Conference on Multimodal Interfaces*, pages 423–428, 2002. 2
- [6] R. G. Cinbis, J. J. Verbeek, and C. Schmid. Weakly supervised object localization with multi-fold multiple instance learning. *IEEE TPAMI*, 39(1):189–203, 2017. 2
- [7] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1-2):31–71, 1997. 2
- [8] C. Feichtenhofer, A. Pinz, and R. P. Wildes. Spatiotemporal residual networks for video action recognition. In *NIPS*, pages 3468–3476, 2016. 2
- [9] C. Feichtenhofer, A. Pinz, and R. P. Wildes. Spatiotemporal multiplier networks for video action recognition. In *CVPR*, pages 7445–7454, 2017. 2
- [10] C. Feichtenhofer, A. Pinz, and A. Zisserman. Convolutional two-stream network fusion for video action recognition. In *CVPR*, pages 1933–1941, 2016. 2
- [11] K. Hara, H. Kataoka, and Y. Satoh. Learning spatio-temporal features with 3d residual networks for action recognition. In *ICCV*, pages 3154–3160, 2017. 2, 3, 6
- [12] K. Hara, H. Kataoka, and Y. Satoh. Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In *CVPR*, pages 6546–6555, 2018. 2, 3
- [13] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick. Mask R-CNN. In *ICCV*, pages 2980–2988, 2017. 3, 4
- [14] J. Hoffman, D. Pathak, T. Darrell, and K. Saenko. Detector discovery in the wild: Joint multiple instance and representation learning. In *CVPR*, pages 2883–2891, 2015. 2
- [15] S. Ji, W. Xu, M. Yang, and K. Yu. 3d convolutional neural networks for human action recognition. *IEEE TPAMI*, 35(1):221–231, 2013. 2
- [16] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 2, 3
- [17] O. Krams and N. Kiryati. People detection in top-view fish-eye imaging. In *AVSS*, pages 1–6, 2017. 2
- [18] J. Li, J. Liu, Y. Wong, S. Nishimura, and M. S. Kankanhalli. Self-supervised representation learning using 360° data. In *ACM Multimedia*, pages 998–1006, 2019. 2
- [19] J. Li, Y. Wong, Q. Zhao, and M. S. Kankanhalli. Dual-glance model for deciphering social relationships. In *ICCV*, pages 2669–2678, 2017. 2
- [20] J. Li, Y. Wong, Q. Zhao, and M. S. Kankanhalli. Unsupervised learning of view-invariant action representations. In *NeurIPS*, pages 1262–1272, 2018. 2
- [21] K. Liu, W. Liu, C. Gan, M. Tan, and H. Ma. T-C3D: temporal convolutional 3d network for real-time action recognition. In *AAAI*, pages 7138–7145, 2018. 7
- [22] L. Liu, Z. Pan, and B. Lei. Learning a rotation invariant detector with rotatable bounding box. *arXiv preprint arXiv:1711.09405*, 2017. 2
- [23] P. H. O. Pinheiro and R. Collobert. From image-level to pixel-level labeling with convolutional networks. In *CVPR*, pages 1713–1721, 2015. 2
- [24] J. Redmon, S. K. Divvala, R. B. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *CVPR*, pages 779–788, 2016. 8
- [25] R. Seidel, A. Apitzsch, and G. Hirtz. Omnidetector: With neural networks to bounding boxes. *arXiv preprint arXiv:1805.08503*, 2018. 2
- [26] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *ICCV*, pages 618–626, 2017. 5, 8
- [27] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. In *NIPS*, pages 568–576, 2014. 2
- [28] R. Stiefelhagen, X. Chen, and J. Yang. Capturing interactions in meetings with omnidirectional cameras. *IJDET*, 3(3):34–47, 2005. 2
- [29] M. Tamura, S. Horiguchi, and T. Murakami. Omnidirectional pedestrian detection by rotation invariant training. In *WACV*, pages 1989–1998, 2019. 2
- [30] D. Tran, L. D. Bourdev, R. Fergus, L. Torresani, and M. Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, pages 4489–4497, 2015. 2
- [31] L. Wang, Y. Qiao, and X. Tang. Action recognition with trajectory-pooled deep-convolutional descriptors. In *CVPR*, pages 4305–4314, 2015. 2
- [32] J. Wu, Y. Yu, C. Huang, and K. Yu. Deep multiple instance learning for image classification and auto-annotation. In *CVPR*, pages 3460–3469, 2015. 2
- [33] H. Xu, A. Das, and K. Saenko. R-C3D: region convolutional 3d network for temporal activity detection. In *ICCV*, pages 5794–5803, 2017. 6
- [34] D. Zhang, D. Meng, C. Li, L. Jiang, Q. Zhao, and J. Han. A self-paced multiple-instance learning framework for co-saliency detection. In *ICCV*, pages 594–602, 2015. 2
- [35] Y. Zhou, X. Sun, Z. Zha, and W. Zeng. Mict: Mixed 3d/2d convolutional tube for human action recognition. In *CVPR*, pages 449–458, 2018. 6
- [36] Z. Zhou and M. Zhang. Multi-instance multi-label learning with application to scene classification. In *NIPS*, pages 1609–1616, 2006. 2
- [37] Z. Zhou, M. Zhang, S. Huang, and Y. Li. Multi-instance multi-label learning. *Artificial Intelligence*, 176(1):2291–2320, 2012. 2