ITE Trans. on MTA Vol. 8, No. 4, pp. 269-279 (2020)

Paper

Multiple Human Tracking with Alternately Updating Trajectories and Multi-Frame Action Features

Hitoshi Nishimura (student member)†,††, Kazuyuki Tasaka †, Yasutomo Kawanishi †,††,
Hiroshi Murase †,††

Abstract In this paper, we propose a multiple human tracking method with alternately updating trajectories and multi-frame action features (MHT-MAF). Even though occlusion or motion blur occurs due to the sudden movement of the drone, ID switches are prevented by the stable MAF. In the experiments, we verified the effectiveness of the proposed method using the Okutama-Action dataset. Our code is available online (`https://github.com/hitottiez/mht-paf`).

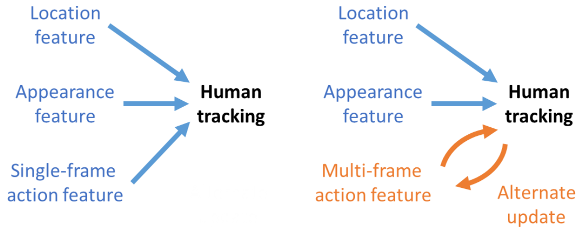
Key words: multiple human tracking, action recognition, drone

1. Introduction

Human trajectory statistics are fundamental information used for marketing, urban development, and sports analysis. To estimate human trajectories, computer-vision-based multiple human tracking is a powerful tool, which detects multiple humans and maintain their identities (IDs) over an image sequence[1]. In places where camera installation is difficult, such as temporary events and festival venues, drones are useful for flexibly capturing scenes.

Human tracking methods are categorized into online tracking methods and offline tracking methods[2]. In this work, we focus on offline tracking methods as they are more accurate than online tracking methods in general because they can use future frame observations when handling the current frame. Most offline tracking methods[3)~8)] are based on the tracking-by-detection fashion due to recent improvements in the accuracy of human detection. Tracking-by-detection solves multiple human tracking problems by data association[3]. Data association matches detection results between consecutive frames based on some cues.

Human tracking methods[3)~8)] use human appearance features and/or human position features as cues. In such methods, when occlusion or motion blur occurs, the human appearance feature and the estimated human position can change dramatically, which can cause frequent ID switches (i.e., that means the target human ID changes to another ID).



(a) Previous methods (b) Proposed method

Fig. 1: Difference between previous methods and the proposed method.

Some human tracking methods[9)~15)] use an action feature as a cue to improve the human tracking accuracy. In such methods, an action feature is extracted from a single frame (Fig. 1(a)).

However, the single-frame action feature (SAF) is unstable because it is extracted from a single frame without considering other frames. The SAF is unstable, especially when occlusion or motion blur occurs due to the sudden movement of the drone. An action feature would be obviously effective if it is extracted from multiple frames, but a multi-frame action feature (MAF) and human tracking are interdependent. Specifically, to extract a MAF for human tracking, the human tracking (trajectory estimation) needs to be done beforehand.

In this paper, we propose a multiple human tracking method with alternately updating trajectories and multi-frame action features (MHT-MAF). Once human tracking is done, MAF is extracted based on the trajectory and used again for human tracking. Furthermore,

Received May 13, 2020; Revised August 5, 2020; Accepted Sepember 11, 2020

† KDDI Research, Inc.
(Saitama, Japan)
†† Graduate School of Informatics, Nagoya University
(Nagoya, Japan)

trajectories and MAF are updated alternately with several steps (Fig. 1(b)). Even though occlusion or motion blur occurs, ID switches are prevented by the stable MAF.

The rest of the paper is organized as follows. First, we review related work in Section 2. Next, we describe conventional multiple human tracking in Section 3. Then, we describe the multiple human tracking using SAF in Section 4. We describe the alternately updating trajectories and MAF in Section 5. In Section 6, we describe the experiments we conducted. Finally, we conclude our work in Section 7. Note that this paper is an extended version of our previous work[16] with the inclusion of an additional method and experiments.

2. Related Work

In this section, we review related work in the areas of human tracking, action recognition, and action detection.

2.1 Human Tracking

In human tracking, multiple humans are detected while maintaining their IDs. In this paper, we focus on offline tracking methods. Zhang *et al.* proposed MCF, which solves human tracking tasks as a minimum-cost flow problem[3]. Berclaz *et al.* reformulated human tracking as a constrained flow optimization in a convex problem[4]. Milan *et al.* proposed a human tracking method that is solved by continuous energy minimization[5]. Bochinski *et al.* proposed a fast and accurate human tracking method by incorporating single object trackers[6]. Maksai *et al.* proposed a method that iteratively builds a rich training set for human tracking[7]. Zhang *et al.* proposed a human tracking and 3D localization method using a drone camera[8]. When occlusion or motion blur occurs, these previous methods are not particularly accurate because they use only human appearance features and/or positions as cues.

2.2 Action Recognition

For action recognition, an action class is estimated based on a given spatio-temporal action position. Many action recognition methods have been proposed[17)~21]. Simonyan *et al.* introduced a two-stream network using RGB and flow images[17]. Our proposed method is based on a two-stream network due to its simplicity. Wang *et al.* proposed TSN, which divides an image into several segments in a temporal domain[18]. Donahue *et al.* introduced LRCN, which performs long-term action recognition using LSTM[19]. Tran *et al.* proposed C3D, which extracts a feature by 3D convolution[20]. Carreira *et al.* proposed I3D, which uses 3D convolution, the parameters of which are based on 2D convolution[21].

2.3 Human Tracking and Action Recognition

There are methods that perform both human tracking and action recognition. Yamaguchi *et al.* proposed an agent-based behavioral model of pedestrians for human tracking[9]. Alahi *et al.* proposed an LSTM model that can learn the general movements of humans and predict human trajectories[10]. Robicquet *et al.* presented a method for predicting human trajectories based on social etiquette[11]. Li *et al.* proposed a human tracking method incorporating action recognition at individual, interaction and group activity levels[12]. Yang *et al.* proposed STAM to obtain an attention for the target human using a drone camera[13]. In these methods, human tracking and action recognition are performed separately; thus, mutual dependencies are avoided. Khamis *et al.* proposed an efficient flow model for joint human tracking and action recognition[14]. Choi *et al.* presented a unified framework for human tracking and group action recognition based on a hierarchical graphical model[15]. These two methods are not particularly accurate when there is occlusion or motion blur because they extract the action feature from only a single frame and do not update it using other frames.

2.4 Action Detection

In action detection tasks, spatio-temporal action positions and action classes are estimated. Many action detection methods have been proposed[19)22)~26]. Action detection tasks can be classified into three categories. (1) Spatial action detection: Gkioxari *et al.* introduced a model based on action tubes constructed from 3D region proposals, CNN features, and SVMs[22]. Lin *et al.* proposed SSAD, which is an end-to-end neural network[23]. (2) Temporal action detection: LRCN performs temporal action detection using LSTM[19]. Shou *et al.* introduced multi-stage CNN that employs 3D CNNs for temporal action detection[24]. (3) Spatio-temporal action detection: Hou *et al.* proposed T-CNN, which is a unified deep neural network that detects actions based on 3D convolution features[25]. Kalogeiton *et al.* proposed an ACT detector that is also a unified deep neural network based on stacking single-frame features[26]. All these methods use action information as a cue for human tracking. However, they do not use a specific human appearance feature that captures human ID as cues.

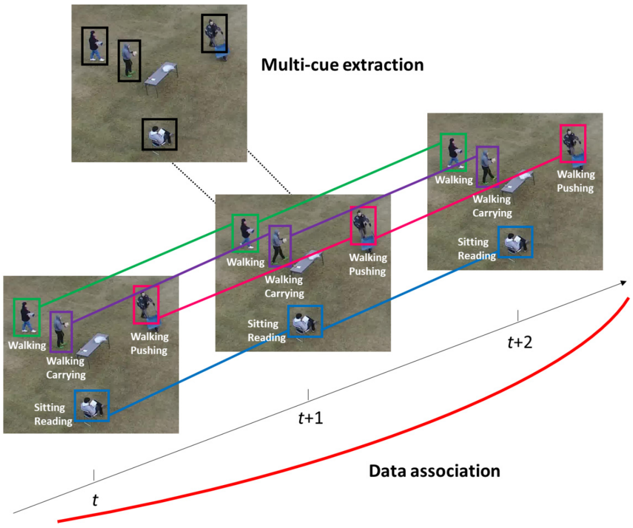


Fig. 2: Multiple human tracking in tracking-by-detection fashion.

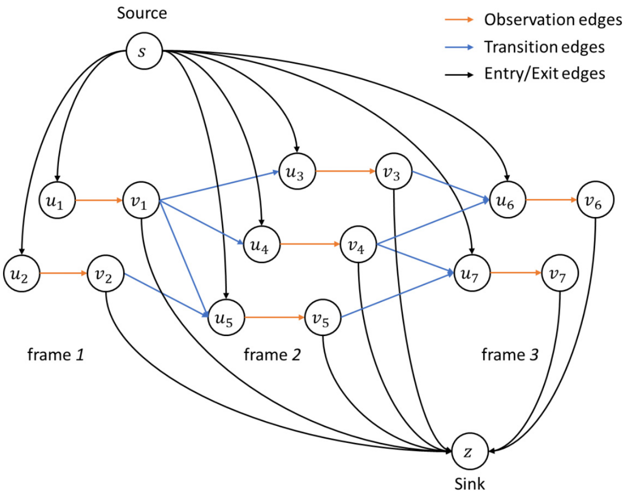


Fig. 3: An example of the cost flow network with 3 frames and 7 observations.

3. Multiple Human Tracking

In this section, we describe the multiple human tracking problem (Section 3. 1) and a general solution to the problem (Section 3. 2).

3. 1 Problem Formulation

Let $Y = \{\mathbf{y}_i\}$ be a set of observations of a human, each of which is a human detection result. The i-th observation is defined as $\mathbf{y}_i = (t_i, \mathbf{b}_i, \mathbf{x}_i)$. t denotes a time step. $\mathbf{b} = (x, y, w, h)$ denotes the bounding box of a human. x and y are the x and y coordinates of the upper left corner of a rectangle, and w and h are its width and height. $\mathbf{x}$ denotes a set of features related to multiple cues. Let $Y_k = (\mathbf{y}_{k_1}, \mathbf{y}_{k_2}, \cdots, \mathbf{y}_{k_{l_k}})$ be the k-th human trajectory. Human tracking is a process for estimate all trajectories $\Omega = \{Y_k\}$, given a sequence of images. Since one human can belong to only one trajectory, we can use the constraint that Y can not overlap with each other as follows:

$$Y_k \cap Y_l = \emptyset, \forall k \neq l. \quad (1)$$

3. 2 Solution

Fig. 2 shows conventional multiple human tracking in tracking-by-detection fashion. First, multiple cues for multiple human tracking are extracted for each frame (Section(1)). Then, data association is performed (Section(2)).

(1) Multi-cue Extraction

For each $\mathbf{y}_i$, two types of cues are often used. From the cues, a location feature ($\mathbf{x}_i^{\text{loc}}$) and an appearance feature ($\mathbf{x}_i^{\text{app}}$), i.e., $\mathbf{x}_i = (\mathbf{x}_i^{\text{loc}}, \mathbf{x}_i^{\text{loc}})$ are extracted. $\mathbf{x}^{\text{loc}} = (x, y, w, h)$ is the bounding box of a human, and it is also used for $\mathbf{b}$. $\mathbf{x}^{\text{app}}$ is an appearance feature.

Location Feature: Each bounding box $\mathbf{x}_i^{\text{loc}}$ and its score x_{sco} are estimated by SSD[27]. The backbone model of the SSD is VGG16[28].

Appearance Feature: The appearance feature $\mathbf{x}_i^{\text{app}}$ captures a human appearance. In the feature space, two features indicate the same human when the distance between their features is small. The distance metric of the feature space is extracted by a Siamese network that has two inputs and one output[29]. The network accepts a pair of human images and outputs a binary indicator of identity. Two backbone models of the Siamese network share the same weights, and each backbone model is WideResNet[30]. In the training phase, while a pair of the same human is annotated as "1", a pair of different humans is annotated as "0". In the inference phase, one of the two backbone models is used to extract the appearance feature.

(2) Data Association

Data association is performed based on the multiple cues described in Section(1.) We formulate data association as a minimum-cost flow problem[3] because that is one of the most typical and intuitive approaches. In a minimum-cost flow problem, data association is modeled as a network, where each node represents an observation and each edge represents a transition between two observations. Source and sink nodes can initialize and terminate trajectories. A solution is obtained by finding the minimum-cost flow in the network.

Fig. 3 shows an example of the cost flow network. For every observation $\mathbf{y}_i$, create two nodes u_i, v_i, create an edge (u_i, v_i) with cost $c(u_i, v_i) = c_{\text{obsv}}(i)$ and flow $f(u_i, v_i) = f_{\text{obsv}}(i)$, an edge (s, u_i) with cost

$c(s,u_i) = c_{\mathrm{entr}}(i)$ and flow $f(s,u_i) = f_{\mathrm{entr}}(i)$, and an edge (v_i,z) with cost $c(v_i,z) = c_{\mathrm{exit}}(i)$ and flow $f(v_i,z) = f_{\mathrm{exit}}(i)$. For every transition from $\mathbf{y}_i$ to $\mathbf{y}_j$, create an edge (v_i,u_j) with cost $c(v_i,u_j) = c_{\mathrm{tran}}(i,j)$ and flow $f(v_i,u_j) = f_{\mathrm{tran}}(i,j)$. The minimum-cost flow problem for human tracking estimates a set of indicator variables F as follows:

$$F = \{(f_{\mathrm{entr}}(i), f_{\mathrm{obsv}}(i), f_{\mathrm{tran}}(i,j), f_{\mathrm{exit}}(i)) \\ \mid \forall i, \forall j, i \neq j, t_i \neq t_j\}, \tag{2}$$

where $f_{\mathrm{entr}}(i)$, $f_{\mathrm{obsv}}(i)$, $f_{\mathrm{tran}}(i,j)$, and $f_{\mathrm{exit}}(i) \in \{0,1\}$.

$c_{\mathrm{obsv}}(i)$, which is the observation cost of the i-th observation, is based on a logit function. A variable of the logit function, probability p, is calculated by a logistic function with the score of the location feature (x_{sco}) as a variable.

$$c_{\mathrm{obsv}}(i) = b - \log\frac{p}{1-p}, \tag{3}$$

$$p = \frac{1}{1+\exp(\alpha+\beta\cdot x_{\mathrm{sco}}(i))}, \tag{4}$$

where b denotes a predefined bias, and α and β are the parameters of the logistic function. $c_{\mathrm{obsv}} \in (-\infty,+\infty)$. In the training phase, α and β are estimated by the Fisher scoring algorithm.

$c_{\mathrm{tran}}(i,j)$, which is the transition cost between the i-th observation and the j-th observation, is based on a logistic function. A variable of the logistic function, q, is calculated by a nonlinear function g.

$$c_{\mathrm{tran}}(i,j) = -\log\frac{1}{1+\exp(q)}, \tag{5}$$

$$q = g(c_{\mathrm{iou}}(i,j), c_{\mathrm{app}}(i,j)), \tag{6}$$

where c_{iou} and c_{app} denote an IoU (Intersection over Union) score and a cosine distance of appearance features, respectively. g is represented by multiple decision trees. $c_{\mathrm{tran}} \in (0,+\infty)$. In the training phase, the parameters of g are estimated by a gradient boosting algorithm[31].

$c_{\mathrm{entr}}(i)$ is the entry cost, which is an initialization cost of the trajectory of the i-th observation. Similarly, $c_{\mathrm{exit}}(i)$ is the exit cost, which is a termination cost of the trajectory of the i-th observation.

F is estimated by minimizing the following objective function with non-overlapping constraints[3]:

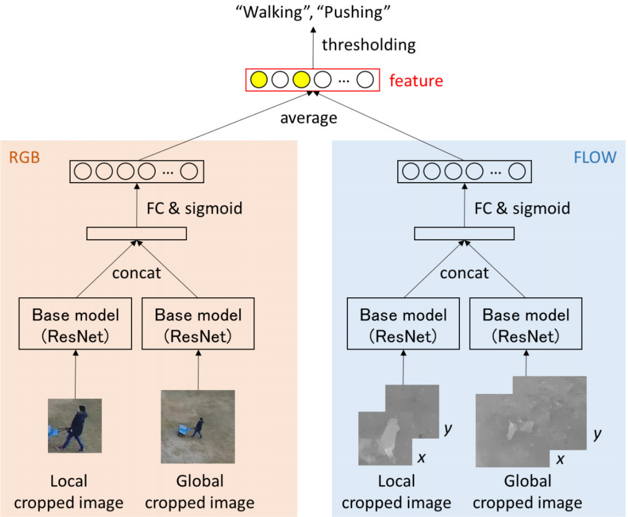


Fig. 4: Single-frame action feature (SAF) extraction model.

$$F^* = \arg\min_F \Bigg[\sum_i c_{\mathrm{entr}}(i) f_{\mathrm{entr}}(i) + \sum_i c_{\mathrm{obsv}}(i) f_{\mathrm{obsv}}(i) \\ + \sum_i \sum_j c_{\mathrm{tran}}(i,j) f_{\mathrm{tran}}(i,j) + \sum_i c_{\mathrm{exit}}(i) f_{\mathrm{exit}}(i) \Bigg]. \\ s.t.\ f_{\mathrm{entr}}(i) + \sum_j f_{\mathrm{tran}}(j,i) = f_{\mathrm{obsv}}(i) \\ = f_{\mathrm{exit}}(i) + \sum_j f_{\mathrm{tran}}(i,j), \forall i \tag{7}$$

The objective function is minimized by the scaling push-relabel method[32].

4. Multiple Human Tracking with Single-Frame Action Feature (SAF)

In the human tracking method described in Section 3, human appearance features and positions are used as cues. When occlusion or motion blur occurs, the human appearance feature and position change dramatically, and ID switches occur frequently. To prevent ID switches, a single-frame action feature (SAF) can serve as an effective cue for data association.

4.1 Single-Frame Action Feature (SAF)

A human region is cropped corresponding to $\mathbf{x}_i^{\mathrm{loc}}$. For each cropped image, a SAF $\mathbf{x}_i^{\mathrm{saf}}$ is extracted. Fig. 4 shows the SAF extraction model. The network is a four-stream neural network.

The network has two modalities, spatial and temporal, and each modality handles the inputs by using a two-stream network[17][18]. While the spatial network utilizes an RGB image, the temporal network utilizes an optical flow image. For optical flow calculation, we used

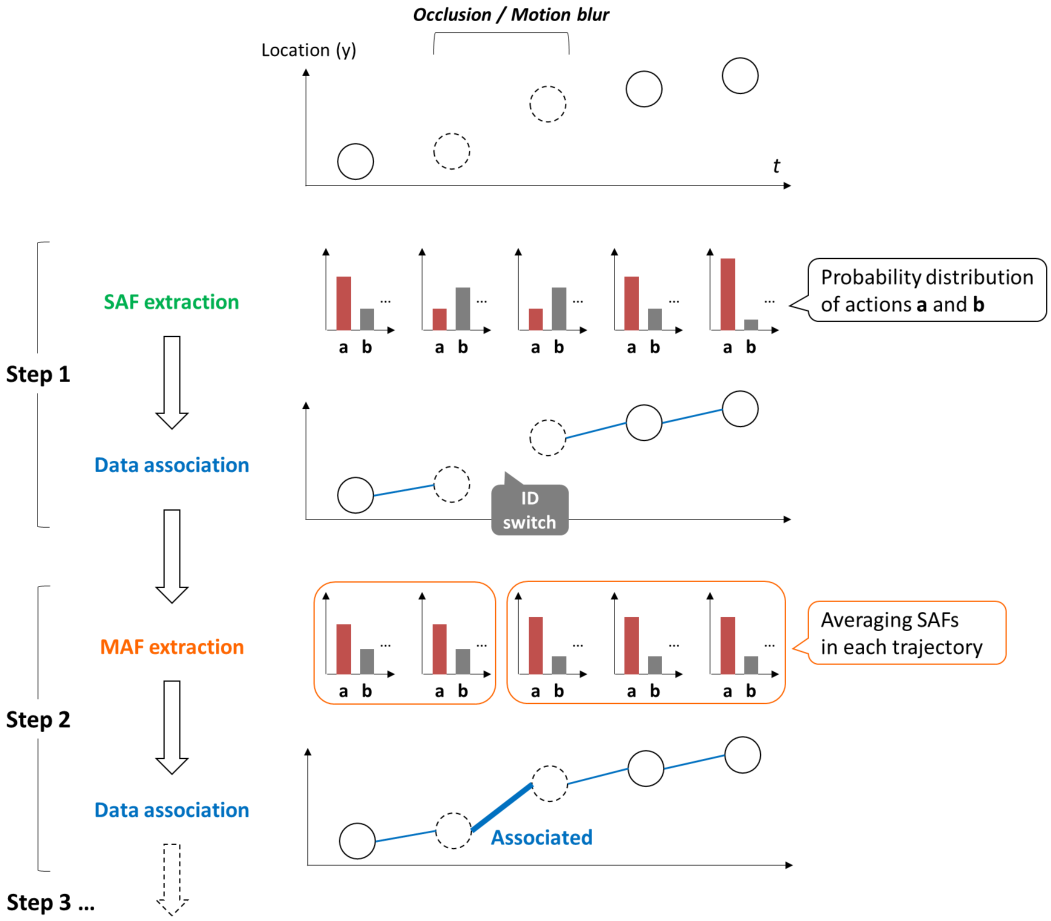


Fig. 5: Alternation of data association and MAF extraction. In this scenario, one human performs the same action **a** across frames, and occlusion or motion blur occurs. The horizontal axis denotes time, and the vertical axis denotes location. The type of line (solid or dotted) figuratively represents the human appearance feature. An action feature (SAF/MAF) for each human is represented as a probability distribution of actions **a** and **b**. Alternation is performed in each step, and ID switching is prevented.

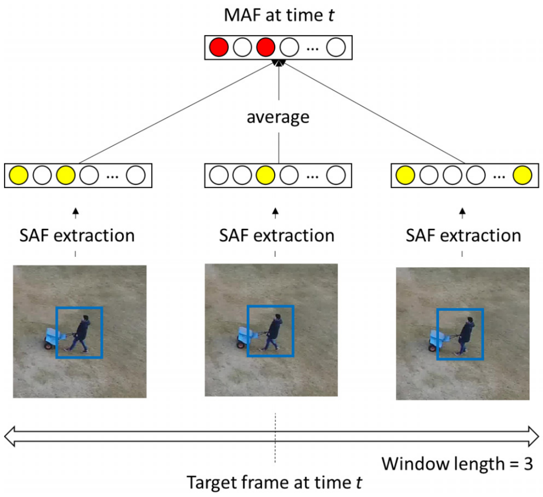


Fig. 6: Multi-frame action feature (MAF) extraction.

TV-L1 optical flow[33], a method that is both fast and accurate. The horizontal and vertical components are used separately. The backbone model of each stream is ResNet101[34].

For each modality, two types of images, i.e., local and global cropped images, are input to the network. The local cropped image is obtained from a square bounding box which fits to the long side of $\mathbf{x}_i^{\mathrm{loc}}$. The global cropped image is obtained based on an expanded bounding box of the local cropped image, taking $\mathbf{x}_i^{\mathrm{loc}}$ as the center. The expansion ratio is set as a predefined parameter μ. The global cropped image introduces the spatial context such as objects and other humans.

The network is trained to recognize multi-label actions such as (walking, carrying). A discriminative action feature can be extracted from the network. The loss function is a binary cross entropy loss for each class.

A SAF $\mathbf{x}_i^{\mathrm{saf}}$ is obtained by averaging the features from spatial and temporal modalities. The feature of each modality is extracted from the layer after a fully connected layer in each modality.

4.2 Transition Cost including SAF

The transition cost $c_{\mathrm{tran}}(i,j)$ is modified to include a SAF. $c_{\mathrm{saf}}(i,j)$, which denotes a cosine distance of a SAF, is added into the nonlinear function g as a variable:

$$r = g(c_{\mathrm{iou}}(i,j), c_{\mathrm{app}}(i,j), c_{\mathrm{saf}}(i,j)). \tag{8}$$

5. Alternate Update of Trajectories and Multi-Frame Action Features (MAF)

The SAF described in Section 4 is unstable because it does not consider other frames. In this paper, using a human tracking result (F^*) and SAF ($\mathbf{x}^{\mathrm{saf}}$), MAF ($\mathbf{x}^{\mathrm{maf}}$) is extracted and used again for human tracking. Furthermore, they are updated alternately over several steps. This alternate update is the key feature of the proposed method.

Fig. 5 shows the alternation of data association and MAF extraction. In this scenario, one human performs the same action **a** across frames, and occlusion or motion blur occurs. The horizontal axis denotes time, and the vertical axis denotes location. The type of line (solid or dotted) figuratively represents a human appearance feature. An action feature (SAF/MAF) for each human is represented as a probability distribution of actions **a** and **b**. When occlusion or motion blur occurs, both the location and appearance feature change remarkably. Alternation is performed in each step.

Step 1: SAF is extracted from each frame, and data association is performed. When occlusion or motion blur occurs, the association fails because the SAF is unstable (action **b** has a higher probability than **a**).

Step 2: Using the trajectories obtained in step 1, MAF extraction is performed. Fig. 6 shows MAF extraction. The red or yellow circles in each vector denote actions, the values of which exceed a predefined threshold. MAF extraction is based on a sliding window. At time t, the MAF $\mathbf{x}(t)^{\mathrm{maf}}$ is calculated as follows:

$$\mathbf{x}(t)^{\mathrm{maf}} = \frac{1}{\lambda} \sum_{t=\lceil t-\lambda/2 \rceil}^{\lfloor t+\lambda/2 \rfloor} \mathbf{x}(t)^{\mathrm{saf}}, \tag{9}$$

where $\lceil\ \rceil$ and $\lfloor\ \rfloor$ denote a ceiling function and a floor function, respectively, and λ is a predefined parameter of the window length. By averaging, MAF is stable (action **a** has a higher probability than **b**) even when the SAF is unstable because of occlusion or motion blur. As a result, the association is correct (action **a** matches action **a**). After step 3, the procedure from step 2 is repeated for the predefined number of steps.

6. Experiments

We conducted human tracking experiments to verify the effectiveness of the proposed human tracking method.

6.1 Dataset

We used the Okutama-Action dataset[35], which is a human action detection dataset based on the aerial view. The dataset is very challenging because it includes significant changes in a human's size and aspect ratio, abrupt camera movement, and dynamic transitions of multi-label actions. The dataset contains 43 videos and was split into training (33 videos) and test data (10 videos). The videos are recorded at 30 FPS, and the total number of images in the dataset is $77,365$. Two drones filmed nine participants from a distance of 10–45 meters and camera angles of 45–90 degrees. The resolution of the images is 4K ($3,840 \times 2,160$). Each bounding box has one or more action labels. Twelve action labels are divided into three categories: human-to-human interactions (handshaking, hugging), human-to-object interactions (reading, drinking, pushing/pulling, carrying, calling), and no-interaction (running, walking, lying, sitting, standing). Multiple actions almost always consist of one no-interaction action and one action from the other two categories.

6.2 Experimental Setting

The human detection model (SSD) was trained using the Okutama-Action dataset for $6,000$ iterations with a learning rate of 10^{-4}. The input size of SSD was 512×512. Note that we used the same human detection results for the previous methods and the proposed method. The appearance feature extraction model (WideResNet) was trained using the MARS dataset[36]. The SAF extraction model was trained using the Okutama-Action dataset for $5,000$ iterations with a learning rate of 10^{-4}. The dropout ratio was set to 0.7. In the data augmentation, random cropping and horizontal/vertical cropping were employed. We empirically set $\mu = 3$, $\lambda = 15$. The observation cost model and the transition cost model were trained using the Okutama-Action dataset. For data association parameters, we empirically set $c_{\mathrm{entr}}(i) = 10$, $c_{\mathrm{exit}}(i) = 10$, and $b = -0.5$.

6.3 Evaluation of Human Tracking

We evaluated the human tracking (Estimating $\mathbf{x}^{\mathrm{loc}}$ and F). The IoU threshold between the ground truth and the estimated bounding box was set to 0.5. For evaluation, we used recall, precision, ID switch (IDs),

Table 1: Human tracking performance.

		Recall (%) ↑	Precision (%) ↑	IDs ↓	MOTA ↑	MOTP ↑
Online	DeepSORT[38]	29.10	70.81	584	17.29	32.85
Offline	MCF[3]	32.42	73.93	597	21.43	32.99
	MHT-SAF	32.13	74.10	558	21.20	**33.01**
	MHT-MAF	**32.48**	**74.24**	**528**	**21.57**	**33.01**

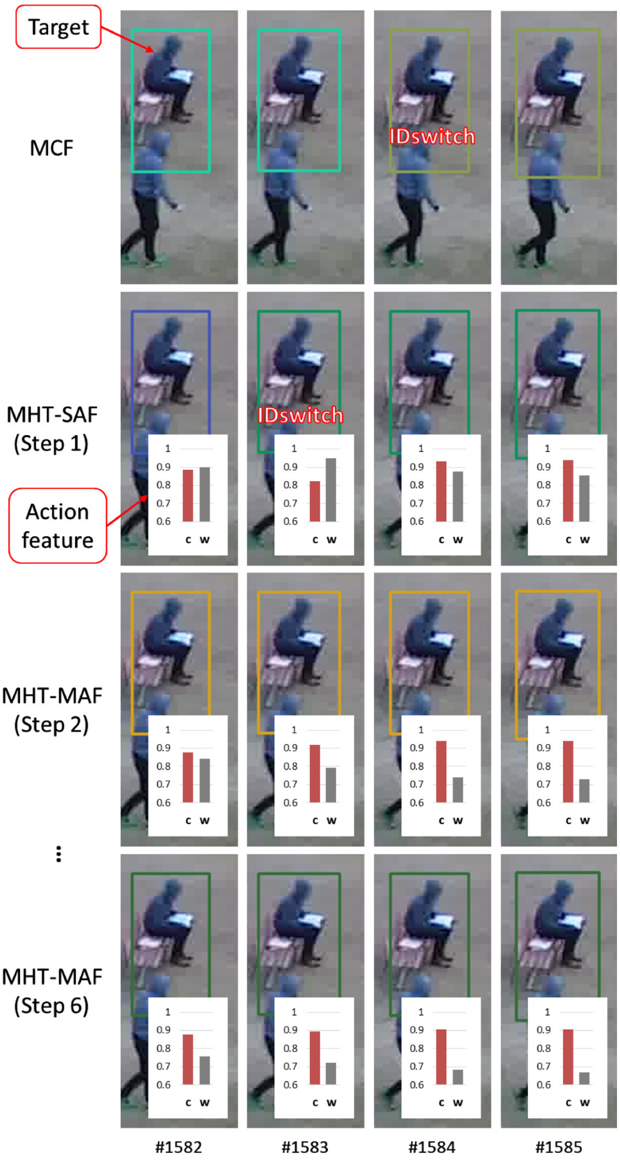


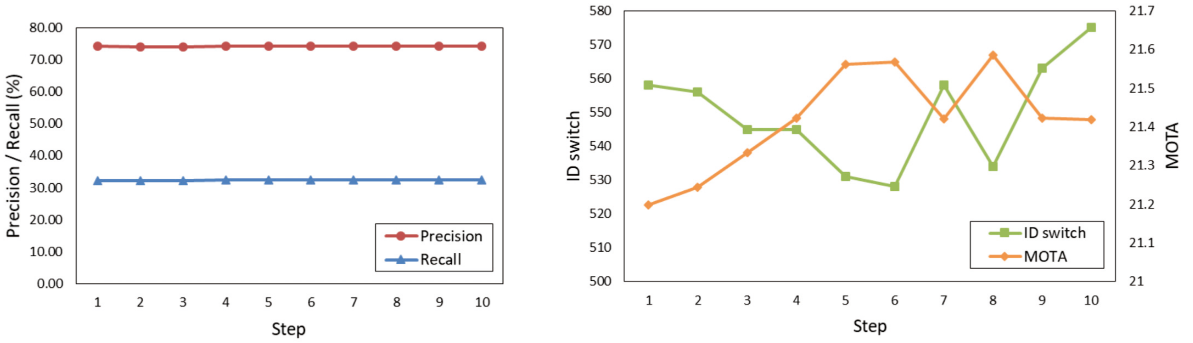
Fig. 7: Example where ID switch is caused by motion blur, but finally the ID switch is prevented by the proposed MHT-MAF.

multiple object tracking accuracy (MOTA), and multiple object tracking precision (MOTP) as metrics[37]. MOTA is a widely used and comprehensive metric that combines three error sources (false negative, ID switch, and false positive).

Table 1 shows the human tracking performance. DeepSORT[38] is an online method that uses location features and appearance features. MCF[3] uses location features and appearance features. MHT-SAF uses location features, appearance features, and SAF without alternations (step 1). MHT-MAF is the proposed method with alternations (step 6). The proposed method maintains almost the same recall (MCF: 32.42; MHT-MAF: 32.48), and almost the same precision (MCF: 73.93; MHT-MAF: 74.24). The number of ID switches decreased by 69 (MCF: 597; MHT-MAF: 528). MOTA improved 0.14 (MCF: 21.43; MHT-MAF: 21.57). MOTP is almost the same (MCF: 32.99; MHT-MAF: 33.01) because MOTP does not consider ID switches.

Fig. 7 shows an example where ID switch occurs because of motion blur, but finally the ID switch is prevented by MHT-MAF. The example was picked up from video 1.2.10. #(number) denotes a frame ID. For each bounding box, an action feature is illustrated, where "c" denotes "carrying" and "w" denotes "walking". In MCF, ID switch occurs at frame 1584 because location and appearance features are unstable. In MHT-SAF (Step 1), frames 1583, 1584, and 1585 are associated via "carrying" and "walking", but frames 1582 and 1583 are not associated. In MHT-MAF (Step 2), frames 1582 and 1583 are estimated as "carrying" due to MAF extraction, then frames 1582, 1583, 1584, and 1585 are associated. Finally, in MHT-MAF (Step 6), all frames 1582, 1583, 1584, and 1585 are stably estimated as "carrying" by repeating MAF extraction.

Next, we evaluate the alternation of human tracking and MAF extraction. Fig. 8(a) shows the recall and precision of each step. In step 1, recall is approximately 74% and precision is approximately 32%. After that, recall and precision remain almost the same. Recall and precision considerably depend on the accuracy of human detection, and it does not change so much with the proposed alternation. Fig. 8(b) shows the number of ID switches and MOTA in each step. The number of ID switches is 558 in step 1. Then, the number of ID switches decreases to 528 in step 6. After that, it increases to 575 in step 10. MOTA has a similar ten-



(a) Recall and Precision.

(b) ID switch and MOTA.

Fig. 8: Performance of human tracking in each step.

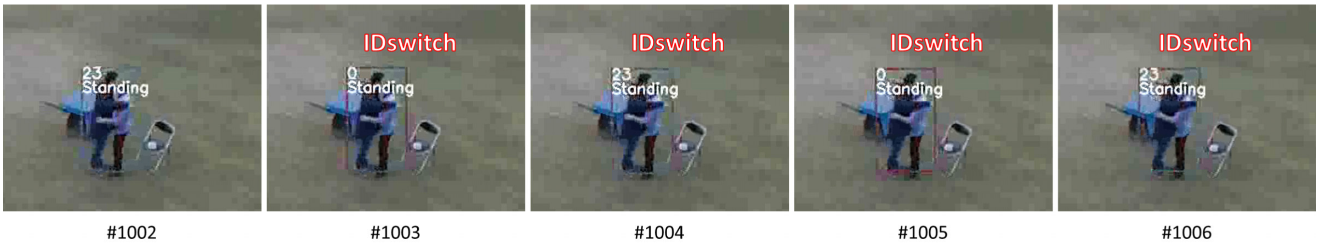


Fig. 9: Example of frequent ID switches because two humans are detected in only one bounding box alternately.

dency as the number of ID switches. The appropriate number of alternations is approximately 5 or 6.

Then, we analyze the scene of frequent ID switches in step 10. Fig. 9 shows an example with two humans who are standing. We picked up 5 frames (frame 1002, ..., 1006) from video 1.2.10. Before those five frames, the humans are tracked as two trajectories (ID:0 and ID:23) for each frame. In the five frames, however, these is only one estimated bounding box (location feature) for each frame. Furthermore, the bounding box includes almost part of the two humans; thus, two trajectories (ID:0 and ID:23) are estimated unstably in the five frames. In the evaluation, the ground truth corresponding to the five estimated bounding boxes (ID:0 and ID:23) is the left man. This phenomenon is a main reason for the frequent ID switches. Preventing the false negatives caused by occlusion is an important issue for future research.

6.4 Evaluation of Action Recognition

We evaluated the accuracy of action recognition, given the ground truth of human tracking. Action recognition is regarded as an estimation problem of $\mathbf{a}_i, \forall i$, where $\mathbf{a}$ denotes a vector of binary values for action classes. When there are any actions, the SAF/MAF values of which are higher than the predefined threshold $\epsilon = 0.4$, then the actions are determined to be recognized. Otherwise, the action of the i-th observation is determined to be “Unknown”. The purpose of the action recognition experiment was to investigate MAF extraction and the global cropped image. The evaluation was performed at the frame level. Table 2 shows the accuracy of action recognition.

Multi-frame-based Action Recognition: For the local cropped image, the accuracy in the case of multi-frame-based recognition is higher than that of single-frame-based recognition (single frame: 42.88; multiple frames: 45.09). For the local+global cropped image, the accuracy in the case of multi-frame-based recognition is higher than that of single-frame-based recognition (single frame: 45.94; multiple frames: 47.80). Therefore, multi-frame-based action recognition, i.e., MAF, is more effective.

Global Cropped Image: Let us compare the local cropped image to the local+global cropped image in single-frame-based action recognition. The accuracy of the local+global cropped image is higher than that of the local cropped image (local: 42.88; local+global: 45.94). For human-to-human interactions and human-to-object interactions, the global cropped image is more effective. These interactions need the global context such as humans or objects for recognition. On the other hand, for no-interaction, the local cropped image is more effective. The no-interaction case needs only human motions for recognition. The average ac-

Table 2: Accuracy of action recognition (%).

	Human to human interactions		Human to object interactions					No-interaction					Average
	handshaking	hugging	reading	drinking	pushing/pulling	carrying	calling	running	walking	lying	sitting	standing	
Single frame (local)	7.78	21.47	57.26	0	55.95	53.57	15.08	43.63	79.97	24.19	76.31	79.37	42.88
Single frame (local+global)	17.11	24.60	61.27	0	64.31	74.13	**17.82**	41.18	85.17	14.33	75.41	75.89	45.94
Multi frames (local)	9.64	17.68	56.75	0	60.88	56.94	13.97	**46.72**	87.19	**26.56**	**81.94**	**82.83**	45.09
Multi frames (local+global)	**18.07**	**24.81**	**61.37**	0	**68.28**	**78.20**	16.76	43.55	**90.73**	15.40	78.04	78.40	**47.80**

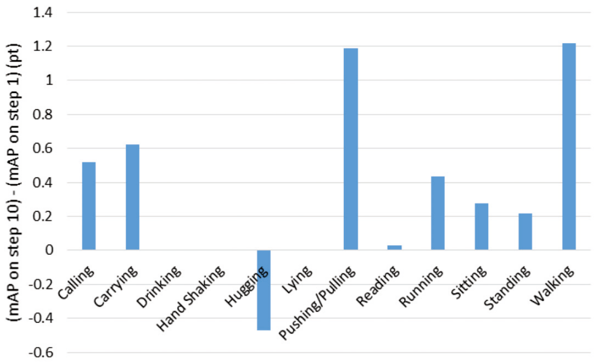


Fig. 10: Difference between mAP when the number of steps is 10 and mAP when the number of steps is 1.

curacy is the highest in the case of the combination of multi-frame-based action recognition and local+global cropped images (47.80).

6. 5 Discussion

The Appropriate Number of Alternations: It may be good to decide the number of alternations based on the type of human action. As described in Section 6. 4, the accuracy of action recognition is improved by using multiple frames. Therefore, if the accuracy of action recognition improves as the number of steps increases, the accuracy of human tracking also improves. We used mean Average Precision (mAP) as an evaluation metric for action recognition. Fig. 10 shows the difference between mAP when the number of steps is 10 and mAP when the number of steps is 1 for each action. The mAP of "pushing/pulling" and "walking" improved by about 1.2 points. As described in Table 2, these two actions have high accuracy to be recognized from a single frame. However, the mAP of "hugging" was reduced by about 0.5 points. As described in Fig. 9, the situation where humans are close to each other causes the frequent ID switches. Thus, it is considered effective to increase the number of iterations when targeting an action that can be easily recognized from a single frame, and to reduce the number of iterations when targeting an action in which multiple humans are close to each other.

Evaluation on Other Dataset: We conducted a human tracking experiment using Drone-Action dataset[39]. The dataset is an action recognition dataset and contains videos recorded from a low-altitude, slow-flying drone. The videos are recorded at 25 FPS. The resolution of the images is HD $(1,920 \times 1,080)$. The experimental setting is the same as described in Section 6. 2. Fig. 11 shows an example of human tracking on Drone-Action dataset. We used the sequence "S8_running_toRight_sideView_HD" because it includes a typical action. In MHT-SAF (Step 1), ID switches occur at frames 40 and 42. In MHT-MAF (Step 2), these ID switches are prevented. "Running" is consistently estimated at all five frames, and it helps the correct association. Table 3 shows the human tracking performance on Drone-Action dataset. The proposed method maintains the same recall (MHT-SAF: 89.68; MHT-MAF: 89.68), and the same precision (MHT-SAF: 83.23; MHT-MAF: 83.23). The number of ID switches decreased by 4 (MHT-SAF: 47; MHT-MAF: 43). MOTA improved 2.58 (MHT-SAF: 41.29; MHT-MAF: 43.87).

7. Conclusion

In this paper, we proposed a multiple human tracking method with alternately updating trajectories and multi-frame action features (MHT-MAF). Even though occlusion or motion blur occurs due to the sudden movement of the drone, MAF is stable and can prevent ID switches. Trajectories and MAF are updated alternately over several steps. In the experiments, we evaluated the proposed method using the Okutama-Action dataset, which consists of aerial view videos. We verified that the number of ID switches decreased by 69 while almost maintaining both recall and precision, and we verified the effectiveness of MAF extraction and the global cropped image. In the future, we will develop a method that prevents false negatives caused by occlusion.

References

1) R.T. Collins, A.J. Lipton, T. Kanade, H. Fujiyoshi, D. Duggins, Y. Tsin, D. Tolliver, N. Enomoto, O. Hasegawa, P. Burt, *et al.*, "A system for video surveillance and monitoring," VSAM final report, pp.1–68, 2000.

2) W. Luo, J. Xing, X. Zhang, X. Zhao, and T.K. Kim, "Mul-

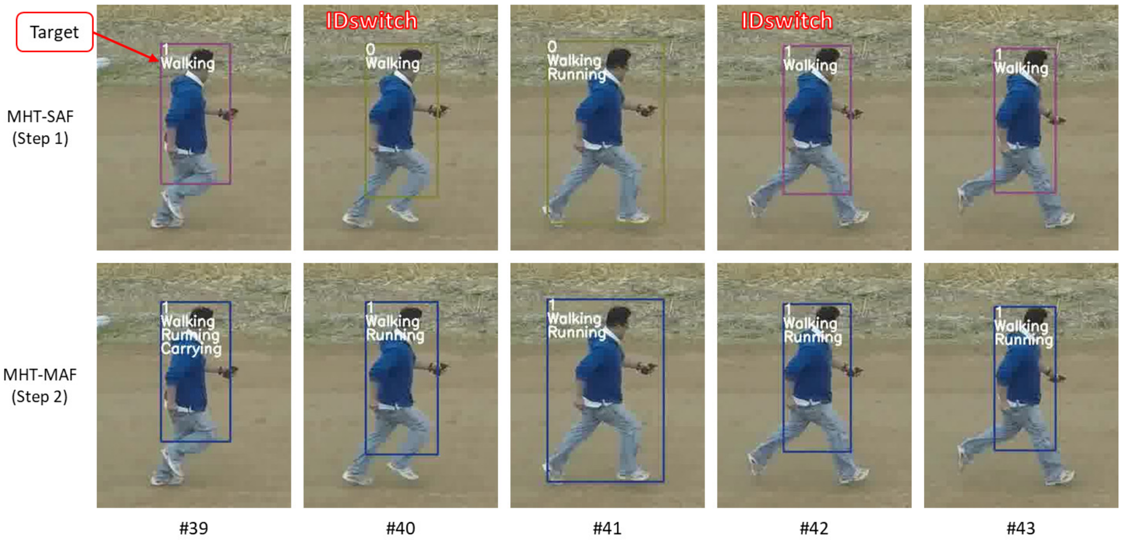


Fig. 11: Example of human tracking on Drone-Action dataset.

Table 3: Human tracking performance on Drone-Action dataset.

	Recall (%) ↑	Precision (%) ↑	IDs ↓	MOTA ↑	MOTP ↑
MHT-SAF	89.68	83.23	47	41.29	33.98
MHT-MAF	89.68	83.23	**43**	**43.87**	33.98

tiple object tracking: A literature review," arXiv preprint arXiv:1409.7618, 2014.
3) L. Zhang, Y. Li, and R. Nevatia, "Global data association for multi-object tracking using network flows," Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp.1–8, 2008.
4) J. Berclaz, F. Fleuret, E. Turetken, and P. Fua, "Multiple object tracking using k-shortest paths optimization," IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), vol.33, no.9, pp.1806–1819, 2011.
5) A. Milan, S. Roth, and K. Schindler, "Continuous energy minimization for multitarget tracking," IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), vol.36, no.1, pp.58–72, 2013.
6) E. Bochinski, T. Senst, and T. Sikora, "Extending iou based multi-object tracking by visual information," Proc. 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), pp.1–6, 2018.
7) A. Maksai and P. Fua, "Eliminating exposure bias and metric mismatch in multiple object tracking," Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp.4639–4648, 2019.
8) H. Zhang, G. Wang, Z. Lei, and J.N. Hwang, "Eye in the sky: Drone-based object tracking and 3D localization," Proc. ACM International Conference on Multimedia (ACMMM), pp.899–907, 2019.
9) K. Yamaguchi, A.C. Berg, L.E. Ortiz, and T.L. Berg, "Who are you with and where are you going?," Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp.1345–1352, 2011.
10) A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social LSTM: Human trajectory prediction in crowded spaces," Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp.961–971, 2016.
11) A. Robicquet, A. Sadeghian, A. Alahi, and S. Savarese, "Learning social etiquette: Human trajectory understanding in crowded scenes," Proc. European Conference on Computer Vision (ECCV), pp.549–565, 2016.
12) W. Li, M.C. Chang, and S. Lyu, "Who did what at where and when: simultaneous multi-person tracking and activity recognition," arXiv preprint arXiv:1807.01253, 2018.
13) F. Yang, S. Sakti, Y. Wu, and S. Nakamura, "A framework for knowing who is doing what in aerial surveillance videos," IEEE Access, vol.7, pp.93315–93325, 2019.
14) S. Khamis, V.I. Morariu, and L.S. Davis, "A flow model for joint action recognition and identity maintenance," Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp.1218–1225, 2012.
15) W. Choi and S. Savarese, "A unified framework for multi-target tracking and collective activity recognition," Proc. European Conference on Computer Vision (ECCV), pp.215–230, 2012.
16) H. Nishimura, K. Tasaka, Y. Kawanishi, and H. Murase, "Multiple human tracking using multi-cues including primitive action features," arXiv preprint arXiv:1909.08171, 2019.
17) K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," Advances in Neural Information Processing Systems (NIPS), pp.568–576, 2014.
18) L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool, "Temporal segment networks: Towards good practices for deep action recognition," Proc. European Conference on Computer Vision (ECCV), pp.20–36, 2016.
19) J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, "Long-term recurrent convolutional networks for visual recognition and description," Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp.2625–2634, 2015.
20) D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3D convolutional networks," Proc. IEEE International Conference on Computer Vision (ICCV), pp.4489–4497, 2015.
21) J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp.6299–6308, 2017.
22) G. Gkioxari and J. Malik, "Finding action tubes," Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp.759–768, 2015.
23) T. Lin, X. Zhao, and Z. Shou, "Single shot temporal action detection," Proc. ACM International Conference on Multimedia (ACMMM), pp.988–996, 2017.
24) Z. Shou, D. Wang, and S.F. Chang, "Temporal action localization in untrimmed videos via multi-stage CNNs," Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp.1049–1058, 2016.
25) R. Hou, C. Chen, and M. Shah, "Tube convolutional neural network (T-CNN) for action detection in videos," Proc. IEEE Inter-

national Conference on Computer Vision (ICCV), pp.5822–5831, 2017.

26) V. Kalogeiton, P. Weinzaepfel, V. Ferrari, and C. Schmid, "Action tubelet detector for spatio-temporal action localization," Proc. IEEE International Conference on Computer Vision (ICCV), pp.4405–4413, 2017.

27) W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.Y. Fu, and A.C. Berg, "SSD: Single shot multibox detector," Proc. European Conference on Computer Vision (ECCV), pp.21–37, 2016.

28) K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556, 2014.

29) N. Wojke and A. Bewley, "Deep cosine metric learning for person re-identification," Proc. IEEE Winter Conference on Applications of Computer Vision (WACV), pp.748–756, 2018.

30) S. Zagoruyko and N. Komodakis, "Wide residual networks," Proc. British Machine Vision Conference (BMVC), pp.1–12, 2016.

31) J.H. Friedman, "Greedy function approximation: a gradient boosting machine," Annals of Statistics, pp.1189–1232, 2001.

32) A.V. Goldberg, "An efficient implementation of a scaling minimum-cost flow algorithm," Journal of Algorithms, vol.22, no.1, pp.1–29, 1997.

33) C. Zach, T. Pock, and H. Bischof, "A duality based approach for realtime TV-L1 optical flow," Proc. Joint Pattern Recognition Symposium, pp.214–223, 2007.

34) K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp.770–778, 2016.

35) M. Barekatain, M. Martí, H.F. Shih, S. Murray, K. Nakayama, Y. Matsuo, and H. Prendinger, "Okutama-action: an aerial view video dataset for concurrent human action detection," Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, pp.28–35, 2017.

36) L. Zheng, Z. Bie, Y. Sun, J. Wang, C. Su, S. Wang, and Q. Tian, "MARS: A video benchmark for large-scale person re-identification," Proc. European Conference on Computer Vision (ECCV), pp.868–884, 2016.

37) K. Bernardin and R. Stiefelhagen, "Evaluating multiple object tracking performance: The CLEAR MOT metrics," EURASIP Journal on Image and Video Processing, vol.2008, no.1, pp.1–12, 2008.

38) N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," Proc. IEEE International Conference on Image Processing (ICIP), pp.3645–3649, 2017.

39) A.G. Perera, Y.W. Law, and J. Chahl, "Drone-Action: An Outdoor Recorded Drone Video Dataset for Action Recognition," Drones, vol.3, no.4, pp.82, 2019.

Hitoshi Nishimura received his B.E. and M.E. degrees in engineering from Kobe University, Japan, in 2013 and 2015, respectively. He joined KDDI Research, Inc. in 2016 and has been engaged in human tracking and action recognition research. He entered Nagoya University in 2018. He is a member of IEICE.

Kazuyuki Tasaka received his B.E. degree from Niihama National College of Technology in 2002. and his M.E. and Ph.D. degree from Nara Institute of Science and Technology in 2004 and 2010, respectively. Since joining KDDI Research, Inc. in 2004, he has worked in the field of network architecture, communication protocols and context recognition. Now, he is a R&D manager in the Media Recognition Laboratory and is a member of IEICE and IPSJ.

Yasutomo Kawanishi received his BEng and MEng degrees in Engineering and a Ph.D. degree in Informatics from Kyoto University, Japan, in 2006, 2008, and 2012, respectively. He became a Post Doctoral Fellow at Kyoto University, Japan in 2012. He moved to Nagoya University, Japan as a Designated Assistant Professor in 2014. Since 2015, he has been an Assistant Professor at Nagoya University, Japan. His research interests are Computer Vision techniques, especially Pedestrian Detection, Tracking, and Retrieval, for surveillance and in-vehicle videos. He received the best paper award from SPC2009, and Young Researcher Award from IEEE ITS Society Nagoya Chapter. He is a member of IEICE, IIEEJ, and IEEE.

Hiroshi Murase received the B.Eng., M.Eng., and Ph.D in engineering from Nagoya University, Japan. From 1980 to 2003 he was a research scientist at the Nippon Telegraph and Telephone Corporation (NTT). He has been a professor of Nagoya University since 2003. He was awarded the IEEE CVPR Best Paper Award in 1994, the IEICE Distinguished Achievement and Contributions Award in 2018. He received Shijyu-hosho (the Medal with Purple Ribbon) in 2012. His research interests include computer vision, pattern recognition, and multimedia information processing. He is a fellow of IEEE, IEICE, and IPSJ.