A. Implementation and Training Details

Backbone network architecture: We use a Cascade Masked RCNN [16] that was pretrained on Cityperson [51] as the localization network to localize the human actor at the start frame. We use X3D-M [10] as the temporal inference backbone to give the final predicted label.
Training details: All the mutual information calculations are implemented on a high-end desktop CPU (Intel Xeon W-2288 CPU), because current version of CUDA does not support histogram operations on GPUs. Our overall model is trained using NVIDIA GeForce 2080Ti GPUs and NVIDIA RTX A5000 GPUs. We use the same initialization as [26]. The initial learning rate is set at 0.1 for training from scratch and 0.05 for initializing with Kinetics pretrained weights. Stochastic Gradient Descent (SGD) is used as the optimizer with 0.0005 weight decay and 0.9 momentum. We use cosine/poly annealing for learning rate decay and multi-class cross entropy loss to constrain the final predictions.
Evaluation: We evaluate our method and other state-of-the-art methods using Top-1 accuracy score, which is the proportion of the correct predictions to all the samples in the evaluation set.

B. Mutual Information

Mutual information is a concept in information theory that essentially measures the amount of information given by one variable when observing another variable. It can also be interpreted as the reduction of the uncertainty of one variable given the other. Mutual information is highly correlated with entropy and joint entropy. The mutual information between image pairs X and Y can be equivalently expressed as:

$$I(X;Y) = H(X) + H(Y) - H(X,Y), \tag{13}$$

where $H(X)$ and $H(Y)$ correspond to the entropy of X and Y, respectively. The entropy quantifies the complexity of all possible outcomes of X or Y. Given $p_X(x)$, $x \in \mathcal{X}$ the probability mass function (PMF) of X, the entropy of X, $H(X)$ can be calculated as:

$$H(X) = -\sum_{x\in\mathcal{X}} p_X(x) \log p_X(x). \tag{14}$$

$H(X,Y)$ is the joint entropy that examines the overall randomness given both X and Y:

$$H(X,Y) = -\sum_{x\in\mathcal{X},y\in\mathcal{Y}} p_{XY}(x,y) \log p_{XY}(x,y), \tag{15}$$

where $p_{XY}(x,y), x \in \mathcal{X}, y \in \mathcal{Y}$ is the joint probability distribution of intensities of pixels associated with X and Y. The joint entropy $H(X,Y)$ is minimized if and only if there is a one-to-one mapping function G such that $p_X(x) = p_Y(G(x)) = p_{XY}(x, G(x))$. It increases when the inherent statistical relationship between X and Y weakens. Therefore, as pixels in X become more distinctive from the counterparts in Y, $H(X,Y)$ gets larger and $I(X;Y)$ gets smaller. Note that, if the image or region pairs X and Y are completely independent from each other, then:

$$\begin{aligned} &H(X,Y) = H(X) + H(Y),\\ &I(X;Y) = 0. \end{aligned} \tag{16}$$

In our case, we use mutual information to obtain and align the region pairs in the temporal domain of a video. Therefore, X and Y are always correlated and $I(X;Y) \neq 0$. Moreover, as we calculate mutual information using probability distribution of discrete pixels, we use sums instead of integrals in Eq . 14 and 15. We use Eq. 14 and 15 to express the mutual information on Eq. 13 using probability distributions. Therefore:

$$I(X;Y) = \sum_{x\in\mathcal{X},y\in\mathcal{Y}} p_{XY}(x,y) \log \frac{p_{XY}(x,y)}{p_X(x)p_Y(y)}. \tag{17}$$

From the equation above, we can see that the mutual information quantifies the dependence between two random variables by measuring the distance between the real joint distribution $p_{XY}(x,y)$ and the distribution under assumption of complete independence of $p_X(x)p_Y(y)$.

Intuitively, as Viola [47] observes, maximizing the mutual information between two images or regions tends to find the most complex overlapping areas (by maximizing the individual entropy) such that at the same time they explain each other well (by minimizing the joint entropy).

The joint mutual information is an extension of mutual information. It measures the statistical relationship between a single variable and a set of other variables. Given one image Y and a set of images X_1, X_2, the joint mutual information is expressed as:

$$I(X_1, X_2; Y) = I(X_1; Y) + I(X_2; Y|X_1). \tag{18}$$

where $I(X_2; Y|X_1)$ is the conditional mutual information that measures the dependence between X_2 and Y when observing X_1.

C. Ablation Experiments

We perform ablation experiments to examine the impact of bin number for histograms to calculate mutual information, reference image size, sliding window stride, searching region and MIS hyperparameters. We randomly pick 30% videos for each action label int UAV-Human and conduct the ablation experiments on this UAV-Human subset. We

Histogram Bin number	Top-1	Reference image Size	Top-1	Sliding Stride	Top-1	Searching area size	Top-1
32	52.3	1.10 ×	53.4	5	52.7	1.25× reference size	52.4
64	52.7	1.25 ×	54.0	10	53.0	1.50× reference size	52.7
128	54.3	1.5 ×	53.7	15	52.8	2.00× reference size	52.5
256	52.7	1.75 ×	52.5	20	51.1	2.50× reference size	52.1

Table 9. Ablation studies on UAV-Human subset in terms of using different bin numbers to calculate mutual information, reference image size (times of the standard size), using different strides for slipping windows, and searching area size. The best performance is achieved while using 128 histogram bins, reference image size 1.25× and sliding stride of 10. The size of the searching area does not affect the overall performance of our method. The top-1 accuracy only varies 0.6% while using different searching area sizes. This demonstrates the robustness of our MITFAS as the larger searching area contains more noises and outliers.

use X3D-M [10] as the temporal inference backbone network. All results are generated by using a sequence of 16 frames with resolution 224×224. All the results are shown in Table. 9.

C.1. Bin Numbers for Histograms

We calculate the mutual information between two images by using their probability distributions. However, there is no exact mathematical model known to precisely calculate the actual probability distributions related to each image. As we mentioned in Eq. 4, we use marginal and joint histograms to approximate the probability distribution. We obtain the joint histogram by binning pairs of pixel values in the two frames. Therefore, bin number is an important hyper parameter for calculating the mutual information. We explore the effects of the number of bins used to generate the joint histogram on the overall performance. We present the results of using different number of bins in Table 9. It shows that the overall accuracy does not monotonically increase as more bins are used and bins number around 128 will result in the best overall performance. It is reasonable because if the histogram is generated with too few bins, then it can not portray the data very well. If too many bins are used, the histogram will not be able to give a good sense of distribution. Therefore, both large and small bin number will lead to bad approximation of the probability distribution, which makes the mutual information calculation less accurate. Moreover, the memory overhead will exponentially grows as more bins are used because the calculation takes the square times of the bin number. To balance the efficiency and accuracy, we use 128 as the bin number for all the experiments in this paper.

C.2. Reference Image Size

Our method needs a reference image without much background information redundancy at the beginning, since we need this inference image as the basis to calculate the mutual information with other frames and eventually obtain a sequence of well aligned regions. However, it is hard to determine how much background information is sufficient enough for aerial recognition as all our videos are captured in the oblique and aerial views with drone cameras. Therefore, we evaluated the impact of different ratio of the background in UAV videos. Let the size of the bounding box generated by the localization network be the standard size. We conduct the experiments on reference images with 4 different sizes (i.e., 1.1 ×, 1.25 ×, 1.5 ×, and 1.75 × of the standard size). As can see in Table . 9, when the reference images is 1.25× of the standard size, we obtain the best performance. Less reference image size makes the model unable to analyze the relationship between the human actor and the surroundings due to less background information. But more background information will bring more noises and outliers, decreasing the overall accuracy.

C.3. Sliding Window Stride

After we obtain the reference image, we use it for MI alignment with the subsequent frame. Here we employ sliding window strategy to find the well-aligned regions that correspond to salient motions in the video. While computing the sliding window, the stride is an important element that needs to be considered since it dramatically effects the overall efficiency. Larger stride means less searching time but decreases the accuracy, as shown in Table 9, stride value at 10 results in the highest accuracy. Therefore, we choose 10 as the sliding window stride for all benchmarks.

C.4. Searching Region

As mentioned in Section 3.4, to reduce the overall mutual information computations, once we compute ω_t^* at time t, we use ω_t^* at $t+1$ to obtain the region $L_{\omega_t^*}(F_{t+1})$. Then, we only search in the searching area which is generated by expanding $L_{\omega_t^*}(F_{t+1})$ by 25% at t+1. Therefore, the size of the searching area is an important hyper parameter for our method. As shown in Table. 9 we conduct experiments with different searching area sizes, $1.25\times, 1.5\times, 2.0\times, 2.5\times$ the size of the $L_{\omega_t^*}(F_{t+1})$, on the UAV-Human subset. Surprisingly, the result shows that the searching area size does not have significant impacts on the overall performance of our method (MITFAS). The top-1 accuracy only varies 0.6% while using different searching area sizes. This demonstrates the robustness of our method, as the larger searching area will contains more noises and outliers. Overall, our MITFAS is robust to outliers and can precisely obtain and align the regions existing salient human motions. Therefore, to reduce the overall training time, we choose the searching area size to be $1.25\times$ the size of $L_{\omega_t^*}(F_{t+1})$ in all other benchmarks in this paper.