Unsupervised and Semi-Supervised Domain Adaptation for Action Recognition from Drones

Jinwoo Choi[1,2] Gaurav Sharma[1] Manmohan Chandraker[1] Jia-Bin Huang[2]
[1]NEC Labs America [2]Virginia Tech

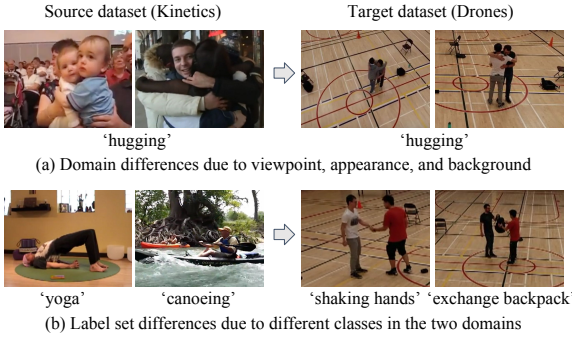
Abstract

We address the problem of human action classification in drone videos. Due to the high cost of capturing and labeling large-scale drone videos with diverse actions, we present unsupervised and semi-supervised domain adaptation approaches that leverage both the existing fully annotated action recognition datasets and unannotated (or only a few annotated) videos from drones. To study the emerging problem of drone-based action recognition, we create a new dataset, NEC-DRONE*, containing* 5,250 *videos to evaluate the task. We tackle both problem settings with 1) same and 2) different action label sets for the source (e.g., Kinectics dataset) and target domains (drone videos). We present a combination of video and instance-based adaptation methods, paired with either a classifier or an embedding-based framework to transfer the knowledge from source to target. Our results show that the proposed adaptation approach substantially improves the performance on these challenging and practical tasks. We further demonstrate the applicability of our method for learning cross-view action recognition on the Charades-Ego dataset. We provide qualitative analysis to understand the behaviors of our approaches.*

1. Introduction

People create large amounts of digital video data recently. Such data comes from many sources e.g., surveillance videos, personal videos, commercial videos, and etc. Many of videos are human-centered. Automatic analysis of videos, e.g., for indexing and searching, is thus an interesting and critical problem. It is also very challenging due to its unconstrained nature and sheer scale. Human action recognition is one of the tasks, in this genre, which has gained substantial attention in recent years [6, 36, 39, 44]. Most of such works have addressed third-person videos while there are some works on egocentric videos as well [11, 40, 54].

Drones are becoming more popular and readily available for purchase in the consumer market. Similar to the existing human-borne camera videos, it is desirable to automatically analyze drone-captured videos. However, drone-captured videos present distinct challenges due to continuous and typical motions, perspectives, and distortions. Thus they are very different from human-borne camera videos (Fig. 1a).



(a) Domain differences due to viewpoint, appearance, and background

(b) Label set differences due to different classes in the two domains

Figure 1: **Action recognition from drone videos**. Transferring knowledge learned from existing action recognition datasets is challenging as they contain mostly third-person videos. We address two challenges, i.e., domain difference (a) due to visual variation as well as (b) due to different label sets, in the two domains.

In this paper, we focus on an unsupervised video domain adaptation setting. We aim to leverage the existing large-scale annotated datasets of third-person videos[1], to help perform action recognition on challenging drone-captured videos. Since acquiring and annotating videos in any new domain is an expensive and time-consuming task, under such domain adaptation settings, we aim to minimize the annotation efforts.

The large domain differences between the *source* domain of third-person videos and the *target* domain of drone-captured videos (Fig. 1a), motivate us to also investigate the case of *semi-supervised* domain adaptation [15]. In the semi-supervised domain adaptation setting, we assume that a limited amount of annotated target data is available during

[1]We refer to existing action recognition datasets such as Kinetics and UCF-101 as third-person datasets while noting that they may contain some other perspective videos, e.g., first person, as well.

978-1-7281-6553-0/20/$31.00 ©2020 IEEE

training in contrast to the unsupervised domain adaptation setting.

In addition to the case where both source and target have the same label sets, we also address the challenging setting where the label sets are *different* (Fig. 1b). To reduce the domain gap between source and target data, we employ a domain classifier and adversarial loss in the both problem settings, i.e., same and different label sets. We use standard cross-entropy loss in the same label set setting, while we use an embedding-based framework in the different label sets case. The input in the latter case is agnostic of the specific class annotations of the training examples. We care only about dis-/similarities between examples, i.e., if they belong to different/same classes irrespective of the particular classes. By employing an embedding-based method, our classifier can generalize to new categories in the target domain.

We also propose to do both full video-based as well as instance-based adaptation. The full video-based method has the merit that exploits correlated context while the instance-based approach is motivated by the argument that focusing on the actor itself is more critical for better performance.

To evaluate the presented methods, we also propose a novel dataset of human actions captured by drones: NEC-DRONE. The NEC-DRONE dataset consists of 5250 videos. We evaluate the proposed method on this challenging dataset and show that we can successfully perform domain adaptation from mostly third-person videos to drone-captured videos. We further evaluate the proposed method on a publicly available Charades-Ego dataset [37]. We show qualitative results on the NEC-DRONE dataset to better understand the behaviors of the methods.

To summarize, we make the following three contributions of this work.

- We introduce a new problem of unsupervised and semi-supervised domain adaptation for action recognition from drones with two settings, i.e., same and different source and target label sets.
- We propose a new dataset, NEC-DRONE, containing 5250 videos for action recognition from drones.
- We explore the problem with thorough experiments and show significant improvements with the proposed method.

2. Related Work

Drone-based video datasets. A few drone-based video datasets have been proposed [3, 26, 30, 56]. However, there is only one dataset for drone-based human action recognition that we are aware of – the OKUTAMA-ACTION dataset [3]. The OKUTAMA-ACTION dataset is an outdoor dataset, and it is 43 minutes total while ours (NEC-DRONE) is 256 minutes. The number of actors is 9 vs. 19 actors (ours), and actions are 12 vs. 16 actions (ours). To the best of our knowledge, the proposed dataset is the largest drone-captured dataset for human action recognition.

Action recognition. After the success of deep networks in the image domain, many works have addressed action recognition in videos [2, 4, 6, 9, 12, 14, 18, 19, 20, 36, 38, 39, 43, 44, 49, 50, 51, 52]. This is in contrast to the earlier handcrafted features [47, 48].

Most of these methods use third-person videos to train their models. In this work, we show that such third-person models do not accurately transfer to novel domains. We propose methods to make models to generalize better using domain adaptation, utilizing a large amount of annotated third-person data.

Cross-view modeling. Understanding object, scene, and action across different views has drawn attention in computer vision. There have been works on aerial and ground view matching [22, 29], albeit the tasks are not human action recognition. For human actions, recent approaches use multi-stream networks to model first and third person videos jointly [1, 10, 35]. However, most of them require a dataset of *paired* videos across views.

We also want to learn view-invariant representations. However, collecting paired videos across different views such as a drone view, a third-person view, and a first-person view is expensive. Thus, we aim to leverage the existing *labeled* third-person videos while using only *unlabeled* target videos (from drones), for learning representations.

Domain adaptation. Many works have addressed the problem of domain adaptation for the case of image classification [13, 15, 24, 25, 31, 34, 45, 46, 55] and object detection [8, 28, 53]. However, not much work has been done on domain adaptation for video-related tasks. A few approaches deal with an image to video domain adaptation [24, 42]. Our work is different as we are interested in a video to video domain adaptation with the target videos being captured by drones.

There are a few works on video domain adaptation [17, 7]. Similar to them, we also use the basic adversarial learning framework. However, we are also dealing with more challenging problem setting where we have different source and target label sets. We are also different in that we propose to use instance-based domain adaptation as our NEC-DRONE dataset has more significant domain gap.

Open set domain adaptation. Open set domain adaptation is the setting where both source and target datasets have 'unknown' classes, and unseen class examples are all classified together into one 'unknown' category [27, 5, 32]. However, we are interested in classifying the unknown examples in different novel classes in the target domain (e.g., 'exchanging backpack').

3. Approach

Our aim is to do domain adaptation from a source domain where we have class annotated training data $(\mathbf{x}_s, \mathbf{y}_s) \in \mathbf{X}^s \times \mathbf{Y}^s$, where $\mathbf{Y}^s$ is the source label set, and unannotated data or a very limited amount of annotated data from the target domain $(\mathbf{x}_t, \mathbf{y}_t) \in \mathbf{X}^t \times \mathbf{Y}^t$ with $\mathbf{Y}^t$ being the target label set. We address two cases of domain adaptation: (i) when the source and target label sets are the same i.e., $\mathbf{Y}^s = \mathbf{Y}^t$, and (ii) when they are different i.e., $\mathbf{Y}^s \neq \mathbf{Y}^t$. We partition the target annotated data into three parts, the usual train and test sets and a third *support set* $(\mathbf{X}^t_N, \mathbf{Y}^t_N)$. We use the support set only in the case of unsupervised domain adaptation with different source and target label sets, to do k-NN classification in the target domain. We report the target performances on the target test set, which we again stress, has no overlap with the support set.

3.1. Overview of the architecture

Our overall architecture (Figure 2) leverages the advances made in both video representations as well as domain adaptation. The system takes a video with T frames, denoted as $\mathbf{V} = \{\mathbf{v}_1, \mathbf{v}_2 \dots \mathbf{v}_T\}$ where $\mathbf{v}_i \in \mathbb{R}^{h \times w \times c}$ are the height h, width w, and c channel frames, as an input and splits it into small, potentially overlapping, clips $\mathbf{x} = [\mathbf{v}_j, \mathbf{v}_{j+1} \dots \mathbf{v}_{j+L-1}]$ where L is the clip length. Then we pass the clips through a state-of-the-art video CNN, denoted as $\psi(\cdot)$ to obtain feature representations, $\psi(\mathbf{x})$ of the clips. We pass the clip features to a softmax with classification loss or an embedding-based metric learning loss, as well as to a discriminator network with domain adversarial loss. We describe the different cases in the following.

3.2. Same source and target label set

The first case is when the K classes are the same in the source and target domains i.e., $\mathbf{Y}^s = \mathbf{Y}^t$ (Figure 2a). Even in this case, the domain differences are substantial due to the various challenges such as variations in appearance, perspective, motion, etc. In this case, the system learns representations with a combination of cross-entropy loss for classification in the source domain along with the domain adversarial loss, i.e., binary cross-entropy loss, between examples of source and target domains. Formally, denoting the classifier by $f_C(\cdot)$ with parameters θ_c, and the discriminator by $f_D(\cdot)$ with parameters θ_d, we define the losses as,

$$\mathscr{L}_{CE} = -\mathbb{E}_{(\mathbf{x}_s, \mathbf{y}_s) \sim (\mathbf{X}^s, \mathbf{Y}^s)} \sum_{k=1}^{K} y_{s,k} \log f_C(\psi(\mathbf{x}_s)), \quad (1)$$

$$\mathscr{L}_{ADV} = -\mathbb{E}_{\mathbf{x}_s \sim \mathbf{X}^s} \log f_D(\psi(\mathbf{x}_s)) - \mathbb{E}_{\mathbf{x}_t \sim \mathbf{X}^t} \log(1 - f_D(\psi(\mathbf{x}_t))). \quad (2)$$

The optimization problem is then given by,

$$\mathscr{L}(\theta_f, \theta_c, \theta_d) = \mathscr{L}_{CE}(\theta_f, \theta_c) - \lambda \mathscr{L}_{ADV}(\theta_f, \theta_d),$$
$$(\theta^*_f, \theta^*_c) = \arg\min_{\theta_f, \theta_c} \mathscr{L}(\theta^*_d), \ \theta^*_d = \arg\max_{\theta_d} \mathscr{L}(\theta^*_f, \theta^*_c). \quad (3)$$

where, θ_f are the feature extractor parameters of ψ, and λ is a hyper-parameter for the trade-off between the cross-entropy and the domain adversarial losses. We mark optimal parameters θ with a symbol $*$ in a superscript.

The optimization learns a classifier by minimizing the classification loss, a discriminator by minimizing the adversarial loss and a feature extractor by minimizing the classification loss and maximizing adversarial loss, to learn domain invariant and discriminative representations. We use the gradient reversal layer [13] for adversarial training.

Semi-supervised adaptation. We also evaluate *semi-supervised* domain adaptation, where, in addition to the unlabeled target examples, some annotated target examples are available for training as well. We use the target *annotated* examples with cross-entropy loss, and the target *unannotated* examples with domain adversarial loss only.

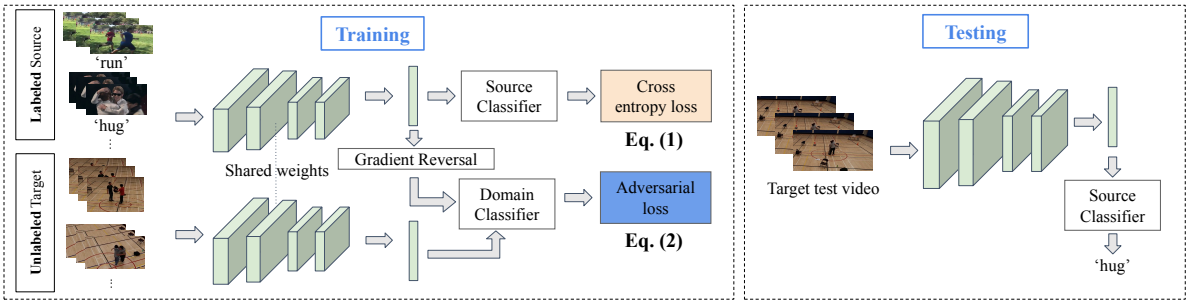
3.3. Different source and target label sets

In the second case, the domain differences are due to the difference in labels sets i.e., $\mathbf{Y}^s \neq \mathbf{Y}^t$ (Figure 2b) as well as the variations such as appearance, perspective, motion, etc. The source and target label sets could be different with some or potentially no overlap. In this case, we propose to learn embeddings of the videos which are agnostic of the specific classes but are aware of being similar (when examples come from the same class) or dissimilar (when they come from different classes). To do this we use a standard metric learning loss, i.e., the triplet loss [33], which takes a triplet of examples $(\mathbf{x}_a, \mathbf{x}_p, \mathbf{x}_n)$ with $\mathbf{x}_a$ being the anchor and $\mathbf{x}_p, \mathbf{x}_n$ being the positive (same class as anchor) and negative (different class than the anchor) examples respectively. In the embedding space, the triplet loss forces the smaller distance between the anchor and the positive example by a margin of δ, than the distance between the anchor and the negative example. Formally the loss and optimization problem are given as,

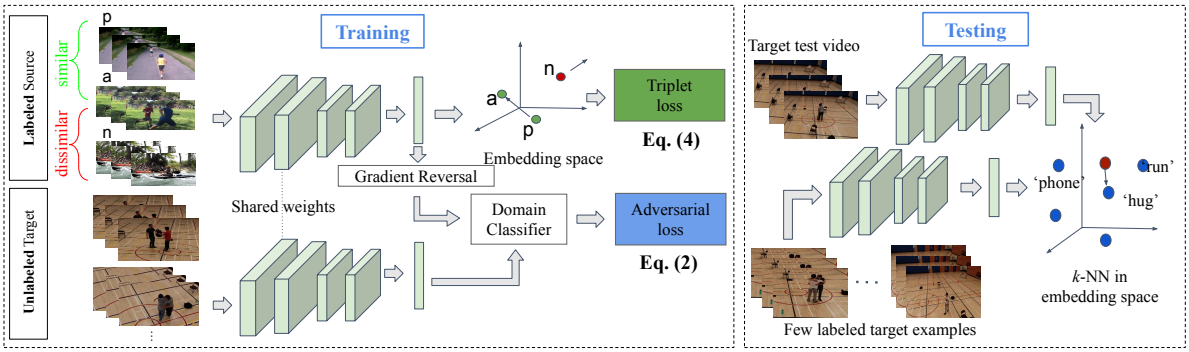
$$\mathscr{L}_{TRI} = -\mathbb{E}_{(\mathbf{x}_a, \mathbf{x}_p, \mathbf{x}_n)} \max(0, \delta + \|\psi(\mathbf{x}_a) - \psi(\mathbf{x}_p)\|^2 - \|\psi(\mathbf{x}_a) - \psi(\mathbf{x}_n)\|^2), \quad (4)$$

$$\mathscr{L}(\theta_f, \theta_d) = \mathscr{L}_{TRI}(\theta_f, \theta_d) - \lambda \mathscr{L}_{ADV}(\theta_f, \theta_d),$$
$$\theta^*_f = \arg\min_{\theta_f} \mathscr{L}(\theta^*_d), \ \theta^*_d = \arg\max_{\theta_d} \mathscr{L}(\theta^*_f). \quad (5)$$

In a minibatch, we sample examples from both the source as well as the target domain. All samples contribute to minimizing the adversarial loss, while the samples from the source domain construct the triplet examples and contribute



(a) Section 3.2. **Same** source and target label sets



(b) Section 3.3. **Different** source and target label sets

Figure 2: **Overview of the proposed domain adaptation method.** Our system takes a video as an input and splits it into small clips. We pass these clips through a video CNN. (a) In the same source and target label set setting, the clips features are input to a softmax with classification loss as well as to a discriminator network with domain adversarial loss. At testing time, the system takes a video as an input, split into multiple clips, pass the clips into the trained CNN to extract features. The system then predicts labels with the source classification layer. (b) In a different source and label sets setting, the clip features are input to an embedding based metric learning loss, as well as to a discriminator network with domain adversarial loss. At testing time, the system takes a video as an input, split into multiple clips, pass the clips into the trained CNN to extract features. The system requires few labeled target examples at test time (a support set) to perform k-NN classification.

to minimizing the triplet loss.

Once we have trained the network, the system classifies query examples, by first, obtaining the embeddings from a forward pass of the base CNN, and then performing k-NN-based classification in the embedding space, using the target support set $(\mathbf{X}_N^t, \mathbf{Y}_N^t)$.

Semi-supervised adaptation. We also evaluate semi-supervised setting in a different source, and target label sets setting, similar to the same label sets setting. The two differences are, (i) we use the cross-entropy loss for target classes as well, and (ii) we do not use the support set, as now the system can directly do target class classification.

3.4. Video-based and instance-based adaptation

Since we are interested in human actions, the discriminative visual regions in the frames are expected to be around humans. We could expect that focusing on the humans in frames would give better performance by eliminating noise from the background. On the other hand, the background might contain correlated elements which could potentially contribute to better recognition. Since both the human foreground as well as the background have potential merits, we propose to do both 'video-based' and 'instance-based' adaptation. In the video-based adaptation we give the full clip as the input to the system, while for the instance-based case, we first perform human detection using a state-of-the-art pre-trained human detector [16] and then feed only the human spatio-temporal tube (i.e., a clip made by cropping out human from every frame) as an input the the system.

We independently train video-based and instance-based domain adaptation models. During testing, we perform late-fusion of the two predictions from video-based and instance-based models. We empirically show that both have advantages, especially when some amount of target annotated data is available (semi-supervised setting), and their combination consistently improves over either of them alone.

4. NEC-Drone Dataset

We propose a new dataset, NEC-Drone, of videos taken from drones for the task of domain adaptation from third-person videos to drone videos. Figure 3 shows some

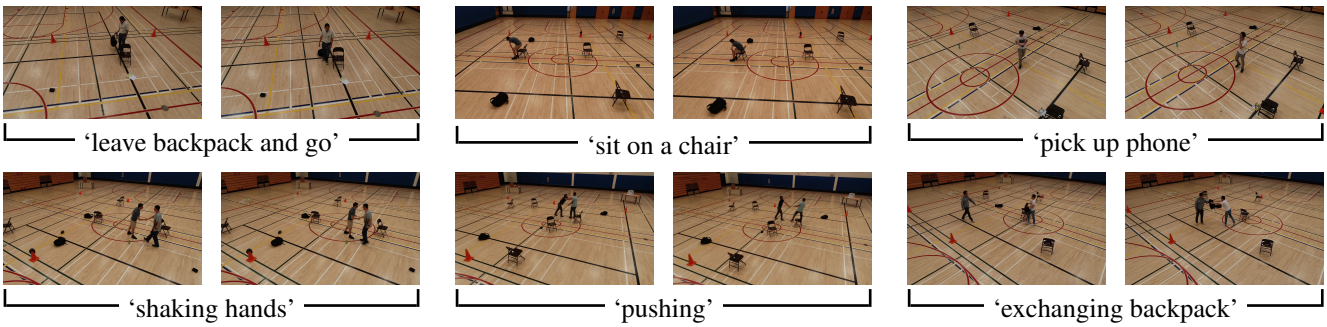


Figure 3: **Sample frames from the NEC-DRONE dataset**. We show two close-by frames per video. The first row shows single person actions, while the second row shows two person actions. Best viewed on screen, with zoom and color.

examples. We collected the dataset inside a school gym with 19 actors acting out their interpretations of 16 predefined actions multiple times. The actions performed by the actors are in an unconstrained manner without any close supervision.

The actions are both single as well as two-person actions. The partial motivation of defining the actions was to keep surveillance scenarios in mind, e.g., two people getting together and exchanging a backpack could be an interesting event to tag, or retrieve. There are 10 single person actions, i.e., walk, run, jump, pick up a backpack and go, leave a backpack and go, sit on a chair, talk on a mobile phone, drink water from a bottle, throw something, pick up a small object, and 6 two-person actions, i.e., shake hands, push a person, hug, exchange a backpack, walk toward each other and stay, stand together leave.

We recorded each action instance by two drones simultaneously flown in an unconstrained manner by relatively new pilots. The videos in the dataset are from varied perspectives with the drones flying at varying distances and heights from the actors. The drones used were 'DJI Phantom 4.0 pro v2' and the videos were recorded at 30 fps at a resolution of 1920×1080 pixels. We manually annotated the actions of all videos.

Finally we have a total of 5250 videos with a total of more than 460*k* frames. We split the videos into 1188 *train*, 437 *val*, and 454 *test* sets with labels, and 3171 videos without labels. We make sure that the actors in the *train*, *val* and *test* sets are disjoint. We evaluate the performance on the dataset as the mean class accuracy.

The proposed dataset is challenging for the following main reasons. First, *view point* is different from typical action datasets, and it changes heavily over time due to the flying drone. Second, due to the *continuous* and often *erratic drone motion*, the videos have jitters and motion blur. Third, often the person(s) of interest are *not centered* and are relatively *small*. To the best of our knowledge, the NEC-DRONE is the largest drone dataset for human action recognition. We plan to release it publicly upon acceptance of the paper.

Table 1: Nearest neighbor *test* results (without learning any parameters) on the UCF-101 and the NEC-DRONE dataset with pre-trained I3D features.

Dataset	UCF-101 (vid.)	NEC-DRONE (vid.)	NEC-DRONE (inst.)
Acc(%)	72.3	8.2	10.8

In Table 1, we show a significantly larger *domain gap* between Kinetics and the NEC-DRONE dataset, compared to the domain gap between Kinetics and UCF-101. We perform a nearest neighbor classification on the UCF-101 [41] and NEC-DRONE datasets. Note that we do not learn any parameters. We use ℓ_2 normalized `mixed5c` activations of the I3D network [6] pre-trained on Kinetics as our feature.

Video-based nearest neighbor classifier can achieve 72.3% accuracy on UCF-101 dataset (split 1). However, the video-based nearest neighbor can achieve only 8.2% accuracy on the NEC-DRONE dataset. Using instance-based nearest neighbor, we can achieve 10.8% accuracy. This significant difference is due to the large domain gap between the NEC-DRONE dataset and typical third-person video datasets. Furthermore, the existing third-person video datasets such as UCF-101 and Kinetics have correlated backgrounds for different actions. However, the NEC-DRONE dataset has a similar background for all the actions. Thus the dataset is very challenging, as without capturing the human motion, it is difficult to recognize the different human actions in the NEC-DRONE dataset.

5. Experimental Results

Abbreviations. We use the following abbreviations. DA: domain adaptation, UDA: unsupervised domain adaptation, SSDA: semi-supervised domain adaptation, src.: source, tgt.: target, vid.: video-based, inst.: instance-based, sup.: supervised finetuning.

Same label set for source and target. We use the Kinetics [21] dataset as the source dataset and the NEC-DRONE dataset as the target dataset. Since the two datasets do not share the same classes, we subsample the two datasets to obtain similar classes. We choose 13 classes from Kinetics [21] dataset and 7 classes from the NEC-DRONE dataset which have similar actions to construct the source and target datasets. See supplementary for the details.

Different label sets for source and target. When we work with different label sets settings, we use the UCF-101 [41] as our source dataset. UCF-101 dataset is mainly a third-person dataset and contains $13,320$ videos from 101 action classes. The domain gap between the UCF-101 and NEC-DRONE datasets is significant (as we also show quantitatively below) and the label sets of the UCF-101 dataset and NEC-DRONE datasets are entirely disjoint. Hence this makes a challenging and practical domain adaptation setting from third-person videos to drone-captured videos.

For both settings, we use labels for m target examples per class, in addition to the unannotated target and annotated source examples, for semi-supervised adaptation.

Implementation details. We use state-of-the-art I3D network [6] as our base network for feature extraction with $L = 16$ frame clip inputs for drone experiments and $L = 32$ frame clip inputs for Charades-Ego experiments. We attach the domain discriminator to `mixed5c` layer of the I3D network. We use a 4 layer MLP for domain classifier where the hidden fully connected layers have 4096 units each.

When aligning features at the instance-based, we extract the human tubes by running per frame detectors and making tracks based on the overlaps of the detections in the successive frames. We use a Mask R-CNN [16] pre-trained on MS-COCO dataset [23] for person detection.

We set $\lambda = 1.0$ for the gradient reversal layer [13], $\delta = 0.5$ for the margin parameter of the triplet loss and the embeddings. We use a batch size of 10 and sample mini-batches as follows. In the case of triplet loss only, 7 out of 10 examples are from anchor class, and the rest 3 are from different classes. In the case of triplet loss with unsupervised domain adaptation, 5 (3 same class, 2 different classes) out of 10 examples are source examples and rest 5 are target examples. We use SGD optimizer with the momentum of 0.9. For the source pre-training and semi-supervised finetuning, we use an initial learning rate of 10^{-4}, and for the unsupervised domain adaptation training, we use an initial learning rate of 10^{-6}. We reduce the learning rate by $1/10$ after 5 epochs.

5.1. Quantitative evaluation on NEC-Drone

Same source and target label sets. We first perform an ablation study of video-based DA and instance-based DA and their combination with different number m of target annotated examples per class used during training. We also include the results without any domain adaptation. Since it is an ablation study, we perform experiments on the *val* set. The column where $m = 0$ is the unsupervised domain adaptation setting while the columns where $m > 0$ are semi-supervised domain adaptation settings.

Table 2: Action recognition accuracies (%) on the NEC-DRONE dataset (*val* set) in the **same source and target label sets** case. m is the number of target annotated examples per class used while training. As a reference, the full target supervised I3D performance is 76.7%.

Method	$m=0$	$m=3$	$m=5$	$m=10$	$m=20$
Inst. no DA	12.6	31.1	35.4	43.2	49.5
Inst. DA	16.5	31.6	39.3	41.3	52.4
Vid. DA	13.6	24.3	35.4	53.9	52.9
Vid. & inst. DA	15.1	32.0	41.8	54.9	58.3

Table 3: Comparison of methods on the NEC-DRONE dataset (*test* set) in the **same source and target label sets** setting, with $m = 5$ target annotated examples per class used in semi-supervised adaptation. The classifier here is the multi-class source classifier.

Method	Training data	Acc (%)	Gain(%)
Fully sup.	*labeled* drone	69.3	N/A
Src. only	Kinetics	13.6	0.0
Vid. DA	Kinetics + *unlabeled* drone	27.2	100.0
Inst. DA	Kinetics + *unlabeled* drone	29.4	116.1
Vid. & inst. DA	Kinetics + *unlabeled* drone	32.0	135.2

The results show the contribution of different adaptation components. The video-based adaptation achieves 13.6% in the unsupervised case, the instance-based achieves 16.5%, while the combination of the two gives 15.1%. The performances rise rapidly as even a small number of annotated examples from the target domain are provided during training. With only $m = 3$ examples the performance of the combined method increases to 32.0% which further increases to $41.8\%, 54.9\%, 58.3\%$ on $m = 5, 10, 20$. The $m = 20$ performance of 58.3% is still far from the full target supervised performance of 76.7%; in the latter case, the average number of examples per class is 80. We also note that the combination of the video-based adaptation with the instance-based adaptation is always greater than either of them indicating complementary information in the two methods.

Table 3, third column, gives the final test performances of the same label set setting for the different methods on the NEC-DRONE dataset for $m = 5$. We see that the video-based adaptation improves the source only classifier from 13.6% to 27.2%, while the instance-based adaptation achieves 29.4%. The combination of both gets the best per-

Table 4: Comparison of methods on the NEC-DRONE dataset (*test* set) in the **different source and target label sets** setting, with, $m = 0$ i.e., unsupervised domain adaptation, and $n = 3$ target examples per class used as a support set at testing. The classifier is nearest neighbor in embedding space.

Method	Training data	Acc (%)	Gain (%)
Fully sup.	*labeled* drone	68.3	N/A
Src. only	UCF101	8.2	0.0
Vid. DA	UCF101 + *unlabeled* drone	10.6	29.2
Inst. DA	UCF101 + *unlabeled* drone	14.3	74.3
Vid. & inst. DA	UCF101 + *unlabeled* drone	14.5	76.8

Table 5: Accuracies (%) on the NEC-DRONE dataset (*test* set) in the **different source and target label sets** case. m is the number of target annotated examples per class used for training. In testing time, we do not use any target examples as a support set. i.e., $n = 0$ setting.

Method	$m = 3$	$m = 5$	$m = 10$	$m = 20$
Inst. DA	15.9	21.6	31.3	34.6
Vid. DA	12.8	18.7	29.1	34.4
Vid. & inst. DA	18.1	22.5	36.1	39.7

formance of 32.0%. This is still quite far from the target fully supervised value of 69.3% indicating that still, a large domain gap exists even after semi-supervised adaptation.

Different source and target label sets. Table 4 shows the results of the different methods for the case of different source and target label sets. Here, we are using no target annotated examples for training ($m = 0$). But we are using $n = 3$ target examples per class at testing as a support set. The task is harder as we use a larger number of classes (all 16 classes present in the NEC-DRONE dataset) compared to the same source and target label sets case, while we use only 7 classes due to the constraint of finding similar classes. The full target supervised accuracy, in this case, is 68.3% compared to 69.3% of the former.

The trends among the methods are similar to the previous case of the same source and target label sets. The source only classifier performs very poorly at 8.2%, cf. 6.25% for a random chance for this 16 class case. The contrast is much higher in this case compared to the previous as (i) it is a harder setting where completely new classes are predicted, and (ii) in general embedding-based methods perform lower than cross entropy-based 1-of-C class classifiers. Compared to the source only classifier, the video-based method improves performance by 29.2% relatively, while the instance-based method improves by 74.3% relatively. The combination of the two further improves 76.8% relatively.

Table 5 gives the semi-supervised domain adaptation results for the setting when the source and target label sets are different. We show the results for the different number m of target annotated examples per class, used during training. The trend is similar to the Table 2. With a small number of annotated examples used during training, we can improve the performance compared to the unsupervised domain adaptation (14.5% for $m = 0$ vs. 39.7% for $m = 20$).

Table 6: Comparison of methods on the Charades-Ego dataset (first person *test* set). Note that for the semi-supervised domain adaptation, we use x% of the target training data with labels and use the rest of the target training data without labels for training.

Method	Back-bone	Pair sup.	Train	Test	% of anno. tgt	mAP (%)
[31]	ResNet-152	✓	3rd + 1st	1st	pair sup.	20.0
Src. only	I3D	×	3rd	1st	0	16.6
UDA	I3D	×	3rd + 1st	1st	0	17.9
SSDA	I3D	×	3rd + 1st	1st	10	20.4
SSDA	I3D	×	3rd + 1st	1st	20	21.9
SSDA	I3D	×	3rd + 1st	1st	30	22.8
SSDA	I3D	×	3rd + 1st	1st	40	23.1
Fully sup.	I3D	×	1st	1st	100	25.8

Figure 4: **Effect of DA and inst. DA.** (Top row) Examples misclassified by the method without DA but are correctly classified with DA. (Bottom row) Examples that are misclassified with vid. DA method but are correctly classified with the vid. & inst. DA. Ground truth/incorrect predictions in green/*red*.

5.2. Quantitative evaluation on Charades-Ego

We compare the performance with other methods on a publicly available Charades-Ego dataset [37] in Table 6. Please note that Charades-Ego is a *paired* dataset. Therefore every first person and third person video is paired with its counterpart. Our method does not require paired dataset, thus more general than Actor and Observer [35]. Also note that the reported performance in [35] is invalid because the authors evaluated on the wrong split[2]. Thus, we run the authors' code and report mAP on the valid test set, which is 20.0%. We obtain 17.9% mAP with our video-based *unsupervised* domain adaptation. Using annotated target data improves performance. With only 10% of the labeled tar-

[2] https://github.com/gsig/actor-observer/issues/7

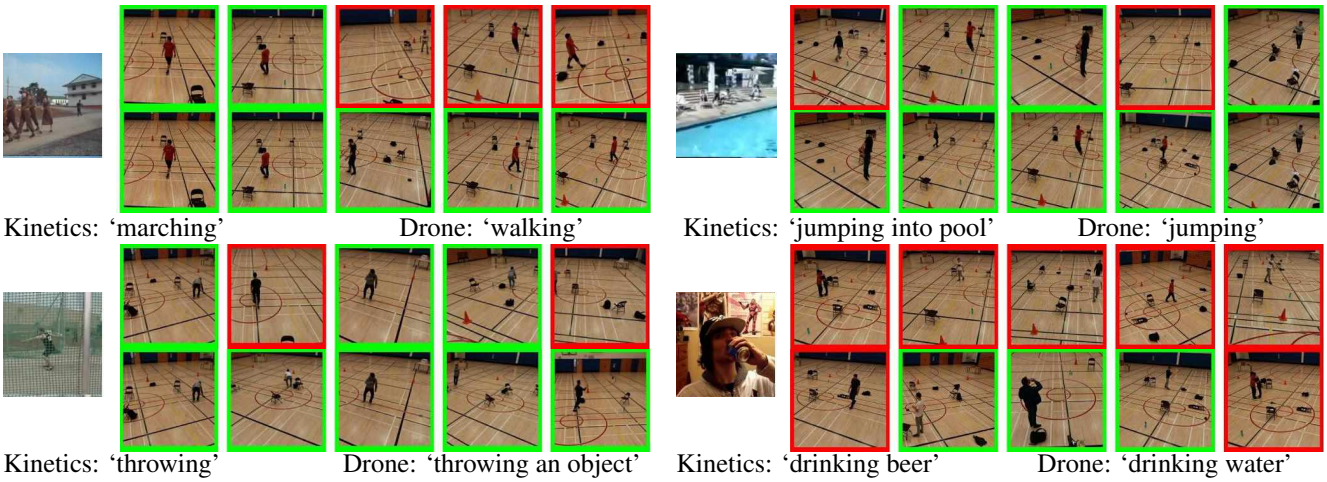


Figure 5: **Cross-domain retrieval results.** For each of the 2×2 blocks, the first column shows a frame of a query video from the Kinetics dataset. The rest of the columns show the top five retrieved videos from the NEC-DRONE dataset. The correct/incorrect category level retrievals are highlighted in green/red. Top row is with the video only model *without* domain adaptation and the bottom row is the same model, but trained *with* domain adaptation. We show the class labels in Kinetics and the corresponding classes from the NEC-DRONE dataset. Best viewed on screen, with zoom and color.

get data, our semi-supervised domain adaptation achieves 20.4% mAP, outperforming the Actor and Observer. Both Actor and Observer and our semi-supervised domain adaptation method use some level of supervision: Actor and Observer uses paired videos for supervision, while our method uses $x\%$ of the target label for supervision. Our semi-supervised domain adaptation method is more general than requiring paired third-person and first-person videos. Therefore, our method is more suitable for action recognition from novel domains where getting paired video dataset is difficult e.g., drone-captured videos.

5.3. Qualitative evaluation on NEC-Drone

We qualitatively demonstrate the contribution of domain adaptation in Figure 4, showing (i) misclassified examples when not using domain adaptation but are correctly classified with domain adaptation (top row), (ii) examples where video only adaptation method fails but combined instance and video-based adaptation method succeeds. We can observe the large domain gaps in terms of the perspective and the area occupied by the actor in the example frames. While the typical third-person videos have a direct perspective and almost centered actor as the major content of the video, the videos in the proposed dataset have challenging perspectives and can also be taken from far. The proposed method addresses these domain gaps and improves performance.

With our embedding-based method, we also obtain a common space where we can compare videos from the source and target domains. To demonstrate it, we show the cross-domain action category level retrieval results in Figure 5. Given a query from 'marching' action class of the Kinetics (first column of the top-left block), we show the top nearest neighbors from the NEC-DRONE dataset. In each block, the first and second row show the retrieval results without and with domain adaptation respectively. We can observe that the domain adpated model can successfully retrieve 'walking' class videos from the NEC-DRONE dataset despite a huge domain gap. Without domain adaptation, the retrievals contain more irrelevant videos from other classes.

6. Conclusion

We addressed the task of human action recognition from drones in the setting where we do not have any labeled examples of drone dataset, or we have only a few labeled examples. We further explored a more challenging setting where the source and the target label sets are different. To deal with this challenging setting, we proposed to use metric learning loss and unsupervised domain adaptation along with instance-level action recognition.

Since a challenging large dataset of drone videos for human action recognition did not exist, we collected 5250 high-resolution videos from two drones with 16 predefined single person and two-person actions. We empirically showed that a large domain gap exists between third-person video datasets and the NEC-DRONE dataset. We will release the dataset upon acceptance to the community.

Our work is among the first to show encouraging domain adaptation results on challenging video domains. However, we also show that we are still far from the fully supervised classifier performances in the target domain of drone videos, and hence, there is much room for improvement.

Acknowledgement. This work was supported in part by NSF under Grant No. 1755785.

References

[1] Shervin Ardeshir and Ali Borji. Integrating egocentric videos in top-view surveillance videos: Joint identification and temporal alignment. In *ECCV*, 2018. 2

[2] Nicolas Ballas, Li Yao, Chris Pal, and Aaron Courville. Delving deeper into convolutional networks for learning video representations. In *ICLR*, 2016. 2

[3] Mohammadamin Barekatain, Miquel Martí, Hsueh-Fu Shih, Samuel Murray, Kotaro Nakayama, Yutaka Matsuo, and Helmut Prendinger. Okutama-action: An aerial view video dataset for concurrent human action detection. In *1st Joint BMTT-PETS Workshop on Tracking and Surveillance, CVPR*, 2017. 2

[4] H. Bilen, B. Fernando, E. Gavves, A. Vedaldi, and S. Gould. Dynamic image networks for action recognition. In *CVPR*, 2016. 2

[5] Pau Panareda Busto, Ahsan Iqbal, and Juergen Gall. Open set domain adaptation for image and action recognition. *TPAMI*, 2018. 2

[6] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, 2017. 1, 2, 5, 6

[7] Min-Hung Chen, Zsolt Kira, Ghassan AlRegib, Jaekwon Woo, Ruxin Chen, and Jian Zheng. Temporal attentive alignment for large-scale video domain adaptation. In *ICCV*, 2019. 2

[8] Yuhua Chen, Wen Li, Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Domain adaptive faster r-cnn for object detection in the wild. In *CVPR*, 2018. 2

[9] Jeff Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *CVPR*, 2015. 2

[10] Chenyou Fan, Jangwon Lee, Mingze Xu, Krishna Kumar Singh, Yong Jae Lee, David J Crandall, and Michael S Ryoo. Identifying first-person camera wearers in third-person videos. In *CVPR*, 2017. 2

[11] Alireza Fathi, Ali Farhadi, and James M Rehg. Understanding egocentric activities. In *ICCV*, 2011. 1

[12] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional two-stream network fusion for video action recognition. In *CVPR*, 2016. 2

[13] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *ICML*, 2015. 2, 3, 6

[14] Rohit Girdhar and Deva Ramanan. Attentional pooling for action recognition. In *NeurIPS*, 2017. 2

[15] Boqing Gong, Yuan Shi, Fei Sha, and Kristen Grauman. Geodesic flow kernel for unsupervised domain adaptation. In *CVPR*, 2012. 1, 2

[16] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. In *ICCV*, 2017. 4, 6

[17] Arshad Jamal, Vinay P Namboodiri, Dipti Deodhare, and KS Venkatesh. Deep domain adaptation in action space. In *BMVC*, 2018. 2

[18] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3d convolutional neural networks for human action recognition. *TPAMI*, 35(1):221–231, 2013. 2

[19] Amlan Kar, Nishant Rai, Karan Sikka, and Gaurav Sharma. Adascan: Adaptive scan pooling in deep convolutional neural networks for human action recognition in videos. In *CVPR*, 2017. 2

[20] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *CVPR*, 2014. 2

[21] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 6

[22] Tsung-Yi Lin, Yin Cui, Serge Belongie, and James Hays. Learning deep representations for ground-to-aerial geolocalization. In *CVPR*, 2015. 2

[23] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *ECCV*, 2014. 6

[24] Zelun Luo, Yuliang Zou, Judy Hoffman, and Li F Fei-Fei. Label efficient learning of transferable representations acrosss domains and tasks. In *NeurIPS*, 2017. 2

[25] Saeid Motiian, Quinn Jones, Seyed Iranmanesh, and Gianfranco Doretto. Few-shot adversarial domain adaptation. In *NeurIPS*, 2017. 2

[26] Matthias Mueller, Neil Smith, and Bernard Ghanem. A benchmark and simulator for uav tracking. In *ECCV*, 2016. 2

[27] Pau Panareda Busto and Juergen Gall. Open set domain adaptation. In *ICCV*, 2017. 2

[28] Anant Raj, Vinay Namboodiri, and Tinne Tuytelaars. Subspace alignment based domain adaptation for rcnn detector. In *BMVC*, 2015. 2

[29] Krishna Regmi and Ali Borji. Cross-view image synthesis using conditional gans. In *CVPR*, 2018. 2

[30] Alexandre Robicquet, Amir Sadeghian, Alexandre Alahi, and Silvio Savarese. Learning social etiquette: Human trajectory understanding in crowded scenes. In *ECCV*, 2016. 2

[31] Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In *CVPR*, 2018. 2

[32] Kuniaki Saito, Shohei Yamamoto, Yoshitaka Ushiku, and Tatsuya Harada. Open set domain adaptation by backpropagation. In *The European Conference on Computer Vision (ECCV)*, September 2018. 2

[33] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, 2015. 3

[34] Ozan Sener, Hyun Oh Song, Ashutosh Saxena, and Silvio Savarese. Learning transferrable representations for unsupervised domain adaptation. In *NeurIPS*, 2016. 2

[35] Gunnar Sigurdsson, Abhinav Gupta, Cordelia Schmid, Ali Farhadi, and Karteek Alahari. Actor and observer: Joint modeling of first and third-person videos. In *CVPR*, 2018. 2, 7

[36] Gunnar A Sigurdsson, Santosh Divvala, Ali Farhadi, and Abhinav Gupta. Asynchronous temporal fields for action recognition. In *CVPR*, 2017. 1, 2

[37] Gunnar A Sigurdsson, Abhinav Gupta, Cordelia Schmid, Ali Farhadi, and Karteek Alahari. Charades-ego: A large-scale dataset of paired third and first person videos. *arXiv preprint arXiv:1804.09626*, 2018. 2, 7

[38] Karan Sikka and Gaurav Sharma. Discriminatively trained latent ordinal model for video classification. *TPAMI*, 40(8):1829–1844, 2018. 2

[39] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *NeurIPS*, 2014. 1, 2

[40] Suriya Singh, Chetan Arora, and C. V. Jawahar. First person action recognition using deep learned descriptors. In *CVPR*, 2016. 1

[41] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012. 5, 6

[42] Kevin Tang, Vignesh Ramanathan, Li Fei-Fei, and Daphne Koller. Shifting weights: Adapting object detectors from image to video. In *NeurIPS*, 2012. 2

[43] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, 2015. 2

[44] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *CVPR*, 2017. 1, 2

[45] Eric Tzeng, Judy Hoffman, Trevor Darrell, and Kate Saenko. Simultaneous deep transfer across domains and tasks. In *ICCV*, 2015. 2

[46] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *CVPR*, 2017. 2

[47] Heng Wang, Alexander Kläser, Cordelia Schmid, and Cheng-Lin Liu. Action recognition by dense trajectories. In *CVPR*, 2011. 2

[48] Heng Wang and Cordelia Schmid. Action recognition with improved trajectories. In *ICCV*, 2013. 2

[49] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Val Gool. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*, 2016. 2

[50] Xiaolong Wang, Ali Farhadi, and Abhinav Gupta. Actions˜ transformations. In *CVPR*, 2016. 2

[51] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, 2018. 2

[52] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning for video understanding. In *ECCV*, 2018. 2

[53] Jiaolong Xu, Sebastian Ramos, David Vázquez, and Antonio M López. Domain adaptation of deformable part-based models. *TPAMI*, 36(12):2367–2380, 2014. 2

[54] Takuma Yagi, Karttikeya Mangalam, Ryo Yonetani, and Yoichi Sato. Future person localization in first-person videos. In *CVPR*, 2018. 1

[55] Jing Zhang, Wanqing Li, and Philip Ogunbona. Joint geometrical and statistical alignment for visual domain adaptation. In *CVPR*, 2017. 2

[56] Pengfei Zhu, Longyin Wen, Xiao Bian, Ling Haibin, and Qinghua Hu. Vision meets drones: A challenge. *arXiv preprint arXiv:1804.07437*, 2018. 2