Improved Generalization of Heading Direction Estimation for Aerial Filming Using Semi-Supervised Regression

Wenshan Wang[1], Aayush Ahuja[1], Yanfu Zhang[2], Rogerio Bonatti[1] and Sebastian Scherer[1]

***Abstract*— In the task of Autonomous aerial filming of a moving actor (e.g. a person or a vehicle), it is crucial to have a good heading direction estimation for the actor from the visual input. However, the models obtained in other similar tasks, such as pedestrian collision risk analysis and human-robot interaction, are very difficult to generalize to the aerial filming task, because of the difference in data distributions. Towards improving generalization with less amount of labeled data, this paper presents a semi-supervised algorithm for heading direction estimation problem. We utilize temporal continuity as the unsupervised signal to regularize the model and achieve better generalization ability. This semi-supervised algorithm is applied to both training and testing phases, which increases the testing performance by a large margin. We show that by leveraging unlabeled sequences, the amount of labeled data required can be significantly reduced. We also discuss several important details on improving the performance by balancing labeled and unlabeled loss, and making good combinations. Experimental results show that our approach robustly outputs the heading direction for different types of actor. The aesthetic value of the video is also improved in the aerial filming task.**

I. INTRODUCTION

Aerial filming is popular with both professional and amateur film makers due to its capability of composing viewpoints that are not feasible using traditional hand-held cameras. With recent advances in state estimation and control technology for multi-rotor vehicles, consumer-level drones became easier to operate; even amateur pilots can easily use them for static landscape filming. However, aerial filming for moving actors in action scenes is still a difficult task for drone pilots and requires considerable expertise. There is increasing demand for an autonomous cinematographer; one that can track the actor and capture exciting moments without demanding attention and effort from a human operator [1].

When filming a moving actor with an autonomous drone, heading direction estimation (HDE) plays a central role, both in terms of motion planning and scene aesthetics. With accurate heading information, the planner can make a better forecast of the actor's future movement, enabling smoother motion, and more reliable visual tracking. In addition, by knowing the actor's heading the drone can execute different types of shots (*e.g* front, back, side shots), depending on the user's artistic objectives.

Estimating the heading of people and objects is also a problem within many other applications, such as pedestrian collision risk analysis [2], human-robot interaction [3] and activity forecasting [4]. However, models obtained for other tasks do not easily generalize to the aerial filming task, due to a mismatch in the types of images from datasets coming from each application. Aerial images tend to have large variations in terms of angles, scale, brightness, and blurriness of the actor (Figure 4). In addition, when the trained model is deployed on a drone, the input actor images used for the HDE module result from an imperfect object detection module, increasing the mismatch between existing datasets [5][6] and the data seen in practice.

Fig. 1: The HDE task. Given the images, we want to predict the heading direction as shown by the red arrows.

Despite the success of deep learning methods, the generalization problem is always one of the major stumbling blocks for applying those methods to real robotic problems, since they rely heavily on labeled data. This is a problem not only because labeling is a laborious task, but also because the labeled data could be biased, resulting in a poor generalization to other applications. Since there are many different types of filming targets such as persons and vehicles, decreasing the amount of labeled data required could be extremely helpful for applying the methods across object types.

This paper presents a semi-supervised algorithm for HDE problem. We utilize temporal continuity as the unsupervised signal to regularize the model and achieve better generalization ability. This semi-supervised algorithm is applied to both training and testing phases, which increase the testing performance by a large margin. We show that by leveraging unlabeled sequences, the amount of labeled data required can be significantly reduced. We also discuss several important details on improving the performance by balancing labeled and unlabeled loss. Experimental results show that our approach robustly outputs the heading direction for different actors in filming task. This results in a better state estimation for the actor in 3D space, also allows us to obtain more visually appealing videos.

In this work we offer three main contributions: 1) We propose a novel semi-supervised approach that uses temporal continuity in sequential data for heading direction estimation problem; 2) We compare our method with baseline

[1]Wenshan Wang, Aayush Ahuja, Rogerio Bonatti and Sebastian Scherer are with the Robotics Institute at Carnegie Mellon University, Pittsburgh, PA 15213, USA {wenshanw,aahuja2,rbonatti,basti}@andrew.cmu.edu

[2]Yanfu Zhang is with Yamaha Motor Co., Ltd., Shizuoka 4370061, Japan zhangya@yamaha-motor.co.jp

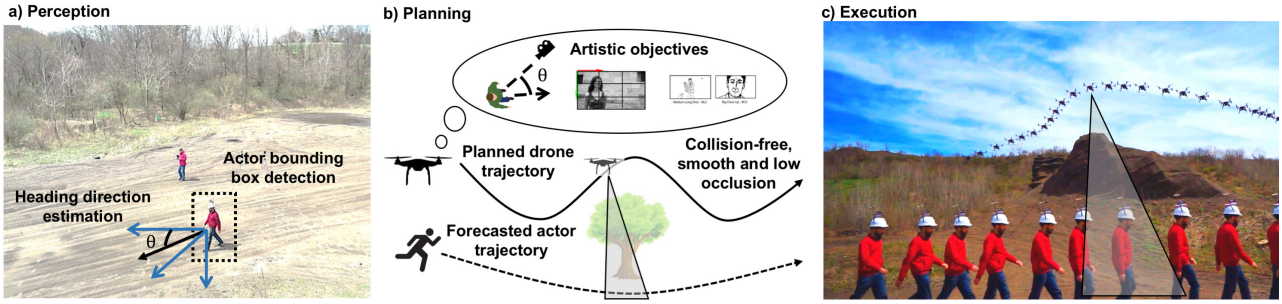

Fig. 2: We propose a system for automatic heading direction estimation (HDE) of a subject, applied for autonomous aerial cinematography. a) The perception system detects a bounding box around the actor, and the HDE network outputs the current heading angle in the world frame. b) The drone reasons about artistic guidelines provided by a user including relative angle to the actor, and calculates a smooth, collision-free trajectory, avoiding occlusion. c) Real-life results validate the system working on an on-board computer, following and tracking the subject online.

approaches, showing that we significantly reduce the amount of labeled data required to train a robust model; and 3) We experimentally verify the robustness of our proposed method, while running on an onboard computer of a drone, following an actor.

II. RELATED WORK

Heading direction estimation is a widely studied problem, in particular focused on humans and cars. One option to tackle the problem is to use inertial and GPS sensors to estimate human's [7], [8] or a car's [9] heading direction. In the context of aerial filming, the target actor generally does not carry extra sensors; thus our emphasis on vision-based solutions in this paper.

Based on a probabilistic framework, Flohr et al. [10] present a joint pedestrian head and body orientation estimation method, in which they design a HOG/linSVM pedestrian detector combined with a Kalman filter. Learning-based methods, however, seem to achieve more robust and generalizable results, being more prevalent in the HDE literature. Most existing learning-based methods use large amounts of labeled data and supervised learning to train a model [11], [12], [13], [14]. However, open datasets [15], [16], [17], [6] generalize poorly to our aerial filming task mainly due to mismatch between image viewpoints, scales, and image blurr. Human key-point detection and 2D pose estimation have also been widely studied [18], [19]. However, such works are focused only on human bodies, and the 3D heading direction cannot be trivially recovered directly from 2D points because the keypoint's depth remains undefined.

Semi-supervised learning (SSL) is also an active research area. Self-training is a commonly used technique for SSL [20]. Weston et al. proposed a graph-based method for semi-supervised classification [21], and recently more related works have been proposed [22], [23], [24] in the area. Most of the existing SSL works are focused on classification problems, which assume that different classes are separated by a low-density area. This assumption is not directly applicable to regression problems.

Temporal continuity is a valid assumption for many robotics sensory data and can be exploited to improve the training of neural networks using SSL. Mobahi et al. [25] developed one of the first approaches to exploit the temporal continuity with deep convolutional neural networks. The authors use video temporal continuity as a pseudo-supervisory signal over the unlabeled data and demonstrate that this additional signal can improve object recognition in videos from the COIL-100 dataset [26]. Other works learn feature representations by exploiting temporal continuity [27], [28], [29], [30], [31]. Zou et al. [27] included the video temporal constraints in an autoencoder framework and learned invariant features accross frames. Wang et al.[31] designed a Siamese-triplet network which can be trained in an unsupervised manner with a large amount of video data. Wang et al. showed that the unsupervised visual representation can achieve competitive performance on various tasks, compared to its ImageNet-supervised counterpart. Srivastava et al. [30] trained a LSTM model in an unsupervised manner by exploiting temporal continuity and demonstrated that this learned representation of video sequences could be used for predicting future frames or improving action recognition accuracy.

Based on the concept of leveraging temporal continuity, we aim to improve the learning of a regression model from a small labeled dataset. The small dataset constraint is common in many robotics applications, where the cost of acquiring and labeling more data is high. We describe our approach in the following section.

III. APPROACH

A. Problem Formulation

To estimate the actor's position and orientation in the world frame, we define HDE and ray-casting modules that process the incoming camera image. We define the pose of the actor as a vector $[x, y, \theta_w]$ on the ground plane. The actor's position $[x, y]$ is inferred by the ray-casting module; it detects a bounding box around the actor at each camera frame, and projects the center of the box's bottom onto the ground plane using the known extrinsic and intrinsic camera parameters. To estimate the θ_w component, we first estimate the actor's heading θ from the image within the bounding box

(Figure 1), and then convert θ to the world frame coordinates using the camera parameters.

The HDE module works by outputting the angle θ in image space. Since θ is ambiguously defined at the frontier between $-\pi$ and π, we define the output of the regressor as two continuous values $[cos(\theta), sin(\theta)]$, therefore avoiding instabilities in the model during training and inference.

We assume we have access to a relatively small labeled dataset $D = \{(x_i, y_i)\}_{i=0}^{n}$, where x_i denotes input image, and $y_i = [cos(\theta_i), sin(\theta_i)]$ denotes the angle label. In addition, we assume access to a large unlabeled sequential dataset $U = \{q_j\}_{j=0}^{m}$, where $q_j = \{x_0, x_1, ..., x_t\}$ is a sequence of temporally-continuous image data. The HDE module's objective is to approximate a function $y = f(x)$, that minimizes the regression loss on the labeled data $\sum_{(x,y)\in D} L_l(x_l, y_l) = \sum_{(x,y)\in D} \|y_i - f(x_i)\|^2$.

B. Defining the loss for temporal continuity

One intuitive way to leverage unlabeled data is to add a constraint that the output of the model should be continuous over a consecutive input sequence. In this sense, the model is trained to jointly minimize the labeled loss L_l and the continuity loss L_u. We minimize the combined loss:

$$\min \sum_{(x_l,y_l)\in L} L_l(x_l, y_l) + \lambda \sum_{q_u \in U} L_u(q_u) \quad (1)$$

The labeled loss could be the mean squared error on labeled data. The continuity loss could be defined in many ways. Intuitively, we look at the samples within one sequence. If two samples are close with respect to temporal distance, their outputs should also be close.

$$L_u(q_u) = \sum_{x_1 \in q_u, x_2 \in q_u} S(x_1, x_2) D(x_1, x_2; f) \quad (2)$$

where S is the similarity between two inputs measured by temporal distance, D is the difference between two outputs produced by the network. To minimize the continuity loss, if two samples within the same sequence are close to each other, their output difference should be small.

The similarity could be defined as follows which takes into account the temporal distance (i.e. the number of consecutive points between the two given points) and it's decay over time.

$$S(x_1, x_2) = e^{-\alpha |n_{x_1} - n_{x_2}|} \quad (3)$$

where α controls the decay speed, n_x is the frame number of input x. The output difference between two samples is taken as the Euclidean distance of the output layer.

$$D(x_1, x_2; f) = \|f(x_1) - f(x_2)\|_2 \quad (4)$$

In practice, we add a small threshold that allows a small difference between consecutive samples. The problem with the previous formulation is the introduction of an additional hyper-parameter α. The hyper-parameter needs to be tuned for different datasets which is difficult and time-consuming in practise. Instead, we look for alternate formulations that alleviate the use of additional hyper-parameter and relax this criterion.

The following formulation of the unsupervised loss is based on the idea that close samples should output smaller difference than far away samples. Similar continuity loss is also used in [31] when training an unsupervised feature extractor.

$$L_u(q_u) = \sum_{x_1,x_2,x_3} \max[0, D(x_1, x_2; f) - D(x_1, x_3; f)] \quad (6)$$

where $x_1, x_2, x_3 \in q_u$ and $S(x_1, x_2) > S(x_1, x_3)$. This usually works better than the previous loss because it doesn't contain additional hyper-parameters and extracts more information from the unlabeled data. The loss definition in Equation 2 indicates two temporally close inputs should have similar outputs. While the loss definition in Equation 6 also suggests that farther inputs should have bigger output difference than closer inputs. This could potentially decrease the small oscillation in output sequence.

C. Network structure

In order to use our network on an onboard computer for real-time applications, we utilize a compact convolutional neural network based on MobileNet [32]. The input of the network is a cropped image of the target, outputted by the detection and tracking modules. The cropped image is padded to a square shape and resized to 192 x 192 pixels. After 10 group-wise and point-wise convolutional blocks as described by MobileNet, we add another convolutional layer and a fully connected layer that output two values, representing the cosine and sine values of the angle (Figure 3).

During each training iteration, one shuffled batch of labeled data and one sequence of unlabeled data are passed through the network. The labeled loss and the unlabeled loss are computed and backpropagated through the network.

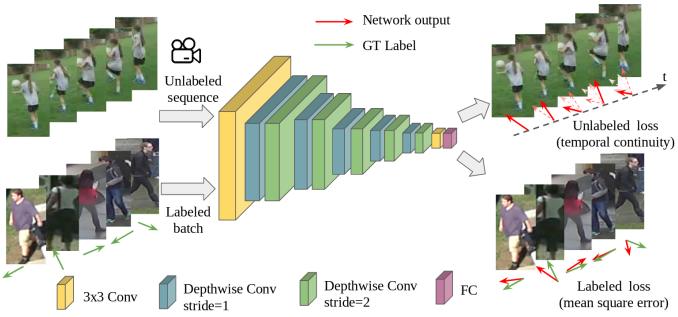

Fig. 3: Our architecture for predicting heading direction. We use mobilenet-based feature extractor followed by a convolutional layer and a fully connected layer to predict angular values. The network is trained using both labeled and continuity loss.

D. Cross-dataset semi-supervised fine-tuning

We use the network architecture explained in the previous section to train on openly accessible datasets. Due to the significant differences in the data distribution between openly available datasets and our drone filming data, the average angle error of the network when tested on the drone filming data is above 30 degree, which is still not good enough for

real-world application. To address the cross-dataset generalization problem, our key insight is to apply the same semi-supervised training idea to finetune the model by employing a few labelled and unlabelled sequences from the drone filming data.

In our drone filming task, the input data being a video is naturally available in a sequential manner, thus allowing the use of the unsupervised loss. We use the insight developed earlier and label a few samples from the video data collected. The HDE model pretrained on openly accessible datasets is finetuned with both labeled loss and unlabeled loss on this collected data. As we will show in the experiment section, using a semi-supervised approach, our model generalizes much better to our drone filming task on unseen sequences and images. Compared with employing only the supervised loss on the drone filming data, it will be shown that generalization is improved while achieving robust and stable performance by employing the unsupervised loss. Our approach makes it feasible for application to other datasets or different kinds of objects since few labelled examples are required and collecting unlabelled videos is easier.

IV. EXPERIMENTS

A. Dataset and baselines

We collected a large number of image sequences from various sources. For the person HDE, we use two surveillance datasets: VIRAT [33] and DukeMCMT [5], and one action classification dataset: UCF101 [34]. However, none of these datasets provides ground truth for HDE. Therefore, in the UCF101 dataset, we manually labeled 453 images for HDE. In the surveillance datasets, we adopted a semi-automatic labelling approach that we first detect the actor in every frame, then compute the ground-truth heading direction based on the derivative of the subject's position over a sequence of consecutive time frames. For the car HDE, two surveillance datasets VIRAT and Ko-PER [35] and one driving dataset PKU-POSS [36] is utilized (Table I).

We find the HDE problem for cars is easier because they are more rigid than human. We use humans as example for the rest of the paper.

TABLE I: Datasets used in this study

dataset	VIRAT	UCF101	DukeMCMT	Ko-PER	PKU-POSS
target	car/person	person	person	car	car
GT	MT*	HL(453)*	MT*	✓	✓
# seqs	650	940	4336	12	-
# imgs	69680	118027	274313	18277	28973

*MT denotes labeling by motion tracking, HL denotes manual labeling.

The data distribution of these datasets is quite different from the data distribution of the drone filming task. The problem is quite challenging since most of the images in our task are very small and blurred (Figure 4).

We compare our approach against two baselines for HDE. The first baseline Vanilla-CNN is a simple convolutional neural network inspired by [12]. The second baseline CNN-GRU implicitly learns temporal continuity using a GRU network inspired by [11]. One drawback for this model is that although it models the temporal continuity implicitly, it needs large number of labeled sequential data for training, which is very expensive to obtain.

Fig. 4: The data for human HDE problem. The data distribution of aerial filming task is very different from the distribution of the open accessible datasets, in terms of image size, blurriness, shot angle and human pose.

We employ three metrics for quantitative evaluation: 1) Mean square error (MSE) between the output $(\cos\theta, \sin\theta)$ and the ground truth $(\hat{\cos}\theta, \hat{\sin}\theta)$. 2) Angular difference (AngleDiff) between the output and the ground truth. 3) Accuracy obtained by counting the percentage of correct outputs, which satisfies AngleDiff$< \pi/8$. We use the third metric, which allows small error, to alleviate the ambiguity in labeling human heading direction.

B. Deceasing labeled data using semi-supervised regression

In this experiment, we train the regression network on the DukeMCMT dataset, which consists of 274k labeled images. Those images are taken from 8 different surveillance cameras. We use the data from 7 of them for training, and one of them for testing (about 50k). We compared our semi-supervised method with supervised one using different number of labeled data and the result is shown in Figure 5. We verify that by utilizing unsupervised loss as Equation (2), the model generalizes better to the validation data than the one with purely supervised loss.

C. Improve generalization ability with semi-supervised fine-tuning

We first train a HDE network on datasets shown in Table I. Those models perform well on the validation set from those datasets. However they generalize poorly to the drone filming task, as shown in row 1-3 of Table II. It is necessary to finetune the model on our data. To minimize the labeling effort, and improve the generalization capbility, we employ semi-supervised method to the finetuning process.

In our aerial filming task, the input data is naturally in consecutive manner. We collect around 50 videos, each

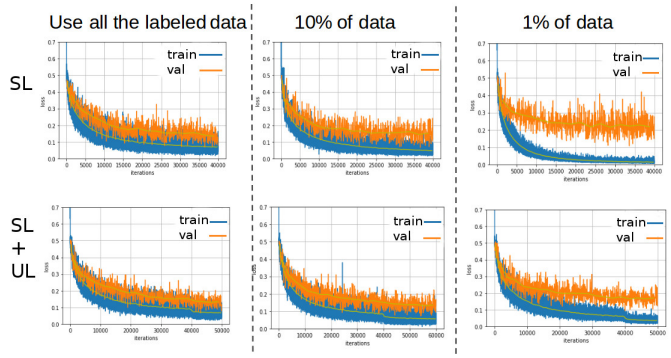

Fig. 5: The top row shows training and validation loss for supervised learning using different number of labeled data. The validation performance drops from 0.13 to 0.22, when decreasing the number of labeled data from 100% to 1%. The bottom row shows results with semi-supervised learning. The validation losses are 0.13, 0.14 and 0.17 respectively for 100%, 10% and 1% labeled data.

contains approximately 500 sequential images. For each video, we manually labeled 6 of those images. The HDE model is finetuned with both labeled loss and continuity loss, same as the training process on the open accessible datasets. We qualitatively and quantitatively show the results of HDE using semi-supervised finetuning in Figure 6 and Table II. The experiment verifies our model could generalize very well to our drone filming task. Compared with the purely supervised learning approach, utilizing unlabeled data improves generalization results and achieves more robust and more stable performance.

We use three metrics to evaluate different methods on aerial filming task (Table II). Vanilla-CNN and CNN-GRU baselines trained on open datasets don't transfer well to drone filming dataset with accuracy below 30%. Our SSL based model trained on open datasets achieves 48.7% accuracy. By finetuning on labelled samples of drone filming, we improve this to 68.1%. Best performance is achieved by finetuning on labelled and unlabeled sequences of the drone filming data with accuracy 72.2%.

TABLE II: Semi-Supervised Finetuning results

Method	MSE loss	AngleDiff (rad)	Accuracy (%)
Vanilla-CNN [12] w/o finetune	0.53	1.12	26.67
CNN-GRU [11] w/o finetune	0.5	1.05	29.33
SSL w/o finetune	0.245	0.649	48.7
SL w finetune	0.146	0.370	68.1
SSL w finetune	**0.113**	**0.359**	**72.2**

D. Implementation details

The λ in Equation (1) is an important hyper-parameter to balance the two losses. These two losses have separate objectives. When λ is small, the continuity loss decreases slowly, while the gap between the training loss and validation loss remains substantial. With large λ, the continuity loss is very low, while both training and validation loss decrease slowly. The network tend to output trivial unchanged outputs to keep the continuity loss low. We achieve best performance using a semi-supervised approach with $\lambda = 0.1$.

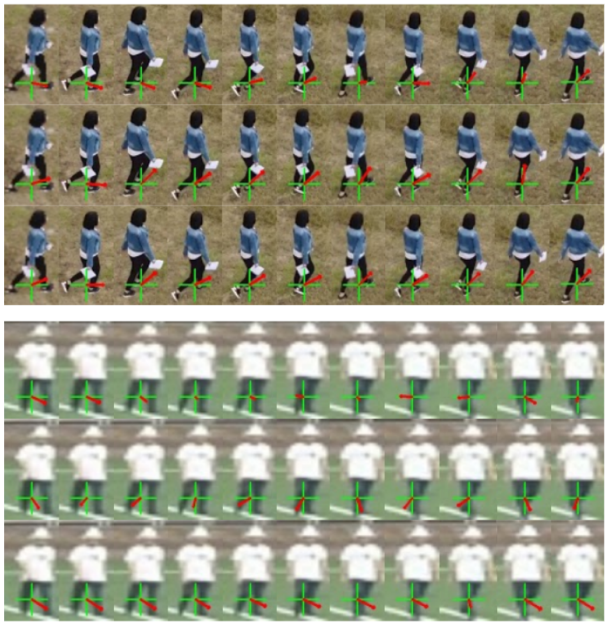

Fig. 6: Three models are tested on the sequential data. Two testing sequences are shown in this figure. The top row of each testing sequence shows the results that directly employ the model trained on other open accessible datasets to the aerial filming task. It generalizes poorly due to the distribution difference. The middle row and bottom row show the results after finetuning the model on the filming data with and without continuity loss respectively. The model using continuity loss for finetuning (bottom row) outputs more accurate and smooth results.

Since collecting unlabeled data is easy, the question is whether having more unlabeled data is always better? If it were true, we could simply collect more data and solve most of the regression problems. Unfortunately, this is not the case. We observe that the unlabeled data improves the performance most only when it is from a similar distribution as the labeled data.

Intuitively speaking, if the distribution of unlabeled data is far from the labeled data, the network would have no idea on the true values of the sequential frames, and it would output continuous values for the sequence in a arbitrary way. To verify this, we evaluated the effect of adding new data in four different settings, and show the validation results on filming data.

1) Add labeled data only: 500 labeled samples (filming dataset).

2) Add unlabeled data only: 1485 unlabeled sequences (filming dataset).

3) Labeled + unlabeled (same distribution): 500 labeled (filming dataset) + 1485 unlabeled sequences (filming dataset).

4) Labeled + unlabeled (different distribution): 500 labeled (UCF101 dataset) + 1485 unlabeled sequences (filming dataset).

The results of these four settings are shown in Figure 7, which plots the validation loss on the filming data. The validation result achieves the best performance when including both the labeled and the unlabeled data from the

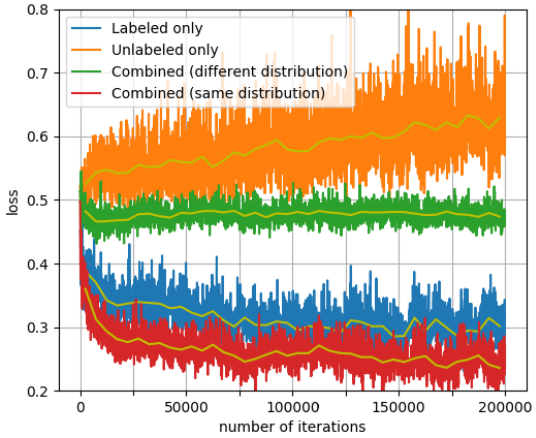

Fig. 7: Validation loss curves on filming data, under four different settings of adding unlabeled data. Orange curve represents adding only unlabeled filming data. Without labeled signal, the validation performance get worse. The green one shows the result of adding labeled and unlabeled data from different distribution. The validation loss drops at first but stay high. The blue curve shows adding only the labeled filming data, which performs well. The red curve shows best result when using both labeled and unlabeled filming data.

filming dataset. This answers the question that the unlabeled sequences alone does not increase the performance. It is important to add a small number of labeled data from similar distribution with the unlabeled sequences.

E. Drone Cinematography using HDE Results

We applied the HDE system to an aerial cinematography platform, and evaluated the real-time HDE results, in addition to improvements in filming quality in comparison with baseline methods for heading estimation. In general, the aerial cinematography platform is broadly divided into vision and planning subsystems. In the vision subsystem, we first detect and track the actor using a monocular camera. We use the bounding box of the tracked actor as an image-space signal to control the camera gimbal, keeping it in the desired screen position. We further adopt the HDE component and a ray-casting approach to estimate the actor's current pose, and a learning-based method [37] to forecast the actor's future poses in the world frame. Details about the hardware and planning subsystem can be found in the work of Bonatti et al. [1].

Due to the limited ways to obtain accurate ground truth in an outdoor environment, we qualitatively evaluate the HDE results as shown in Figure 8. We can see that the HDE system gives consistent and accurate estimations. Before adopting the HDE system, we use an external GPS/compass setup to obtain the actor's pose. Then applying motion forecasting and planning techniques to achieve pre-defined filming patterns. However, this external sensor setup is highly undesired because it significantly constraints the actor's motion and degrades the image aesthetics. In the complementary video, we show that the HDE system can be integrated into the aerial cinematography system and replace the external sensor setup. And Figure 9 shows that replacing the external sensor setup with the HDE system can greatly improve the aesthetics of the images captured by the drone cinematographer.

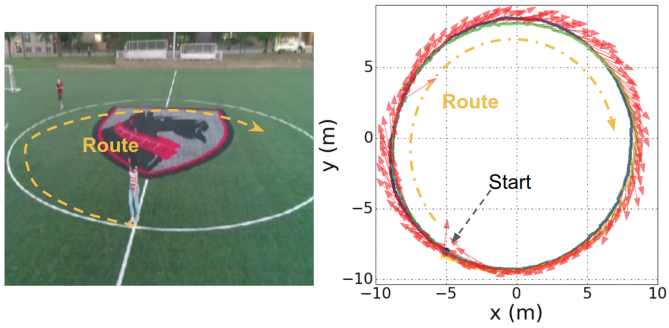

Fig. 8: Qualitative evaluation of the HDE results. The actor walks twice on a circle as shown in the left figure. The actor intentionally keeps the heading to be the same as the tangential direction w.r.t. the circle. In the right figure, red arrows represent the estimated actor heading direction. We can see that the estimated heading direction keeps good consistence and aligns well with tangential direction. With the accurate heading estimation, various shot types, such as side shot or front shot, can be achieved without requiring the actor to wear additional sensor setup.

Fig. 9: Comparison of resulting drone images taken using a compass-GPS setup and visual HDE. The use of a compass and GPS module significantly decrease image aesthetics, and proves to be impractical in most real-life applications for a drone cinematographer. Meanwhile, the visual HDE requires less hardware, is simpler, and allows more complex and constraint-free actor motions.

V. CONCLUSIONS

In this work, we propose a semi-supervised method for HDE problems by leveraging temporal continuity in consecutive unlabeled inputs. We employ the semi-supervised framework to training and finetuning steps, and show that it significantly reduce the amount of labeled data required to train a robust model. We experimentally verify the robustness of our proposed method, while running on an onboard computer of a drone, following an actor. We plan to apply this framework to other mobile robot tasks in the future.

ACKNOWLEDGMENT

Research presented in this paper was funded by Yamaha Motor Co., Ltd. under award #A019969.

REFERENCES

[1] R. Bonatti, Y. Zhang, S. Choudhury, W. Wang, and S. Scherer, "Autonomous drone cinematographer: Using artistic principles to create smooth, safe, occlusion-free trajectories for aerial filming," in *International Symposium on Experimental Robotics (ISER)*, 2018.

[2] R. Tian, L. Li, K. Yang, S. Chien, Y. Chen, and R. Sherony, "Estimation of the vehicle-pedestrian encounter/conflict risk on the road based on tasi 110-car naturalistic driving data collection," in *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*, pp. 623–629, IEEE, 2014.

[3] M. Vázquez, A. Steinfeld, and S. E. Hudson, "Parallel detection of conversational groups of free-standing people and tracking of their lower-body orientation," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pp. 3010–3017, IEEE, 2015.

[4] K. M. Kitani, B. D. Ziebart, J. A. Bagnell, and M. Hebert, "Activity forecasting," in *European Conference on Computer Vision*, pp. 201–214, Springer, 2012.

[5] E. Ristani, F. Solera, R. Zou, R. Cucchiara, and C. Tomasi, "Performance measures and a data set for multi-target, multi-camera tracking," in *European Conference on Computer Vision workshop on Benchmarking Multi-Target Tracking*, 2016.

[6] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *International Journal of Robotics Research (IJRR)*, 2013.

[7] D. Liu, L. Pei, J. Qian, L. Wang, P. Liu, Z. Dong, S. Xie, and W. Wei, "A novel heading estimation algorithm for pedestrian using a smartphone without attitude constraints," in *Ubiquitous Positioning, Indoor Navigation and Location Based Services (UPINLBS), 2016 Fourth International Conference on*, pp. 29–37, IEEE, 2016.

[8] Z. Deng, W. Si, Z. Qu, X. Liu, and Z. Na, "Heading estimation fusing inertial sensors and landmarks for indoor navigation using a smartphone in the pocket," *EURASIP Journal on Wireless Communications and Networking*, vol. 2017, no. 1, p. 160, 2017.

[9] F. P. Vista, D.-J. Lee, and K. T. Chong, "Design of an ekf-ci based sensor fusion for robust heading estimation of marine vehicle," *International Journal of Precision Engineering and Manufacturing*, vol. 16, no. 2, pp. 403–407, 2015.

[10] F. Flohr, M. Dumitru-Guzu, J. F. Kooij, and D. M. Gavrila, "A probabilistic framework for joint pedestrian head and body orientation estimation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 4, pp. 1872–1882, 2015.

[11] P. Liu, W. Liu, and H. Ma, "Weighted sequence loss based spatial-temporal deep learning framework for human body orientation estimation," in *Multimedia and Expo (ICME), 2017 IEEE International Conference on*, pp. 97–102, IEEE, 2017.

[12] J. Choi, B.-J. Lee, and B.-T. Zhang, "Human body orientation estimation using convolutional neural network," *arXiv preprint arXiv:1609.01984*, 2016.

[13] M. Braun, Q. Rao, Y. Wang, and F. Flohr, "Pose-rcnn: Joint object detection and pose estimation using 3d object proposals," in *Intelligent Transportation Systems (ITSC), 2016 IEEE 19th International Conference on*, pp. 1546–1551, IEEE, 2016.

[14] S. Prokudin, P. Gehler, and S. Nowozin, "Deep directional statistics: Pose estimation with uncertainty quantification," *arXiv preprint arXiv:1805.03430*, 2018.

[15] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu, "Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, pp. 1325–1339, jul 2014.

[16] R. Raman, P. K. Sa, B. Majhi, and S. Bakshi, "Direction estimation for pedestrian monitoring system in smart cities: an hmm based approach," *IEEE Access*, vol. 4, pp. 5788–5808, 2016.

[17] W. Liu, Y. Zhang, S. Tang, J. Tang, R. Hong, and J. Li, "Accurate estimation of human body orientation from rgb-d sensors," *IEEE Transactions on Cybernetics*, vol. 43, no. 5, pp. 1442–1452, 2013.

[18] A. Toshev and C. Szegedy, "Deeppose: Human pose estimation via deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1653–1660, 2014.

[19] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *CVPR*, 2017.

[20] D. Yarowsky, "Unsupervised word sense disambiguation rivaling supervised methods," in *Proceedings of the 33rd annual meeting on Association for Computational Linguistics*, pp. 189–196, Association for Computational Linguistics, 1995.

[21] J. Weston, F. Ratle, H. Mobahi, and R. Collobert, "Deep learning via semi-supervised embedding," in *Neural Networks: Tricks of the Trade*, pp. 639–655, Springer, 2012.

[22] E. Hoffer and N. Ailon, "Semi-supervised deep learning by metric embedding," *arXiv preprint arXiv:1611.01449*, 2016.

[23] A. Rasmus, M. Berglund, M. Honkala, H. Valpola, and T. Raiko, "Semi-supervised learning with ladder networks," in *Advances in Neural Information Processing Systems*, pp. 3546–3554, 2015.

[24] Z. Dai, Z. Yang, F. Yang, W. W. Cohen, and R. R. Salakhutdinov, "Good semi-supervised learning that requires a bad gan," in *Advances in Neural Information Processing Systems*, pp. 6513–6523, 2017.

[25] H. Mobahi, R. Collobert, and J. Weston, "Deep learning from temporal coherence in video," in *Proceedings of the 26th Annual International Conference on Machine Learning*, pp. 737–744, ACM, 2009.

[26] S. A. Nene, S. K. Nayar, H. Murase, *et al.*, "Columbia object image library (coil-20)," 1996.

[27] W. Zou, S. Zhu, K. Yu, and A. Y. Ng, "Deep learning of invariant features via simulated fixations in video," in *Advances in neural information processing systems*, pp. 3203–3211, 2012.

[28] R. Goroshin, J. Bruna, J. Tompson, D. Eigen, and Y. LeCun, "Unsupervised learning of spatiotemporally coherent metrics," in *Proceedings of the IEEE international conference on computer vision*, pp. 4086–4093, 2015.

[29] D. Stavens and S. Thrun, "Unsupervised learning of invariant features using video," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pp. 1649–1656, IEEE, 2010.

[30] N. Srivastava, E. Mansimov, and R. Salakhudinov, "Unsupervised learning of video representations using lstms," in *International conference on machine learning*, pp. 843–852, 2015.

[31] X. Wang and A. Gupta, "Unsupervised learning of visual representations using videos," in *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 2794–2802, IEEE Computer Society, 2015.

[32] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[33] S. Oh, A. Hoogs, A. Perera, N. Cuntoor, C.-C. Chen, J. T. Lee, S. Mukherjee, J. Aggarwal, H. Lee, L. Davis, *et al.*, "A large-scale benchmark dataset for event recognition in surveillance video," in *Computer vision and pattern recognition (CVPR), 2011 IEEE conference on*, pp. 3153–3160, IEEE, 2011.

[34] K. Soomro, A. R. Zamir, and M. Shah, "Ucf101: A dataset of 101 human actions classes from videos in the wild," *arXiv preprint arXiv:1212.0402*, 2012.

[35] E. Strigel, D. Meissner, F. Seeliger, B. Wilking, and K. Dietmayer, "The ko-per intersection laserscanner and video dataset," in *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 1900–1901, IEEE, 2014.

[36] C. Wang, Y. Fang, H. Zhao, C. Guo, S. Mita, and H. Zha, "Probabilistic inference for occluded and multiview on-road vehicle detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 1, pp. 215–229, 2016.

[37] Y. Zhang, W. Wang, R. Bonatti, D. Maturana, and S. Scherer, "Integrating kinematics and environment context into deep inverse reinforcement learning for predicting off-road vehicle trajectories," in *Conference on Robot Learning*, 2018.