INTERACTIVE AND INTELLIGENT CAMERA VIEW COMPOSING

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Hao Kang

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

August 2019

Purdue University

West Lafayette, Indiana

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF DISSERTATION APPROVAL

Dr. Bedrich Benes, Chair

Department of Computer Graphics Technology

Dr. Tim McGraw

Department of Computer Graphics Technology

Dr. Yingjie Chen

Department of Computer Graphics Technology

Dr. Haoxiang Li

Wormpex AI Research

Approved by:

Dr. Kathryne Newton

Associate Dean for Graduate Programs, Purdue Polytechnic Institute

ACKNOWLEDGMENTS

I would like to thank my advisor Dr. Bedrich Benes, who gave me numerous excellent advice in both research and life. More importantly, Dr. Benes taught me to see more of the positive side of things, that makes a more optimistic version of me. I also would like to thank my dissertation committee members for the useful discussions and timely feedback, and the HPCG colleges and friends for the strong support. Lastly, I sincerely thank my wife and my family. I cannot get this done without your encouragement and sacrifice. Thank you.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure Page

Figure Page

ABBREVIATIONS

3D	Three-dimensional
AGV	A Good View
AI	Artificial Intelligence
CG	Computer Graphics
DL	Deep Learning
DoF	Degrees of Freedom
EF	Effort
FoV	Field of View
FPV	First-person View
FR	Frustration
GCS	Ground Control Station
GPS	Global Positioning System
GUI	Graphical User Interface
HDI	Human-Drone Interaction
HIT	Human Intelligence Task
HMD	Head-Mounted Display
HRI	Human-Robot Interaction
LSTM	Long-Short Term Memory
MD	Mental Demand
ML	Machine Learning
NASA-TLX	NASA Task Load Index
NMD	Normalized Manhattan Distance
OA	Obstacle Avoidance
OP	Overall Performance

PD	Pysical Demand
RC	Remote Controller
RL	Reinforcement Learning
RoI	Region of Interest
ROS	Robot Operating System
SUS	System Usability Scale
TD	Temporal Demand
VEN	View Evaluation Net

ABSTRACT

Kang, Hao Ph.D., Purdue University, August 2019. Interactive and Intelligent Camera View Composing. Major Professor: Bedrich Benes.

Camera view composing, such as photography, has become inseparable from everyday life. Especially with the development of drone technology, the flying mobile camera is accessible and affordable and has been used to take impressive photos. However, the process of acquiring the desired view requires manual searches and adjustments, which are usually time consuming and tedious. The situation is exacerbated with difficulty in the controlling of a mobile camera that has many Degree of Freedom. It becomes complicated to compose a well-framed view, because experience, timing, and aesthetic are all indispensable. Therefore, professional view composing with a mobile camera is not an easy task for most people. Powered by deep learning, recent breakthroughs in artificial intelligence have enabled machines to perform human-level automation in several tasks (He, Zhang, Ren, & Sun, 2016; Silver et al., 2016). The advances in automatic decision-making and autonomous control have the potential to improve the camera view composing process significantly.

We observe that (a) the human-robot interaction can be more intuitive and natural for photography tasks, and (b) the drone photography tasks can be further automated by learning professional photo taken patterns with data-driven methods. In this work, we present two novel frameworks for drone photography basing on the two observations. First, we demonstrate a multi-touch gesture-controlled gimbaled-drone photography framework-FlyCam. FlyCam abstracts the camera and the drone into a single flying camera object and supports the entire control intuitively on a single mobile device with simple touch gestures. Second, we present

a region-of-interest based, reinforced drone photography framework-Dr^{3}Cam. Our full automation Dr^{3}Cam is built on top of state-of-the-art reinforcement learning research and enables the camera agent to seek for good views and compose visually appealing photos intelligently. Results show that FlyCam can significantly reduce the workload and increase the efficiency in human-robot interaction, while Dr^{3}Cam performs human-level view composing automation for drone photography tasks.

Figure 1.: A Purdue bell tower photo composed autonomously with Dr^{3}Cam.

CHAPTER 1. INTRODUCTION

Camera view composition is vital to obtain informative and visually appealing views in both virtual and real applications. For example, in a virtual environment, the composition is essential to animation and game frames; in practical applications, it is critical in photography and cinematography. However, getting a good camera view usually requires knowledge, experience and sense of aesthetics. Moreover, operating a high Degree of Freedom (DoF) mobile camera, such as a drone, in 3D space is not intuitive due to the *lost in space* pitfall (Christie et al., 2008) and requires training. Hanson and Wernert (1997) proposed a low (2) DoF controller solution to mitigate the high DoF camera control problem in CG. From an aesthetic perspective, the composition of visual elements are commonly based on symmetry, golden ratio, distribution, such as *Rule-of-thirds, Rule-of-odds, Rule-of-space* and *Subframing* (Barnbaum, 2017) shown in Figure 1.1. The principals can be evaluated by algorithms for views (Liu et al., 2010), and further automate the photography with a low DoF mobile camera (Zabarauskas & Cameron, 2014), that allows orientation and position adjustments for simple objectives, to simplify photo acquiring procedure. Similarly, the goal of our study dedicates to reducing high DoF mobile camera control difficulty during the composition process for capturing aesthetic pleasing views.

1.1 "A Good View"

Aiming at improving the mobile camera view composing process, it is necessary to understand what makes a good photo. Despite the rule-breaking masterpieces, general good photos follow empirical composition principles with the consideration of light, color, etc.

Figure 1.1.: The explanation of four basic composition rules (Forbes, 2019). (a) Rule-of-thirds: subdividing the photo into thirds both vertically and horizontally, the sections where dividing lines cross are points of interests. (b) Rule-of-odds: framing the subject with two surrounding objects suggests balance and harmony visually. (c) Rule-of-space: creating negative space that relates to the subject can lead to a sense of motion, activity, or conclusion in the composition. (d) Subframing: framing the target with lines within the composition, thus having a picture in a picture, which can provide emphasis on the subject.

Many existing photography systems follow heuristic photographic composition rules. Zabarauskas and Cameron (2014) summarized several rules that are widely applied basing on the photographic composition book written by Grill and Scanlon (1990):

- *Rule of thirds*, which suggests that the points of interest in the scene should be placed at the intersections (or along) the lines which break the image into horizontal and vertical thirds.
- *No middle rule*, which states that a single subject should not be placed at a vertical middle line of the photograph.
- *No edge rule*, which states that the edges of an ideal frame should not be crossing through the human subjects.
- *Occupancy rule*, which suggests that approximately a third of the image should be occupied by the subject of the photograph (Zabarauskas & Cameron, 2014, p. 1812).

There are similar studies on how to obtain aesthetically delightful views using heuristic principles for a 3D object (Gooch, Reinhard, Moulding, & Shirley, 2001) and human targets (Cavalcanti, Gomes, Meireles, & Guerra, 2006).

However, as the saying by Naoto Fukasawa, *"aesthetics is a beauty that is found by a relationship between things, people and the environment"* - heuristic rules or the "hand-crafted" features usually do not adequately account for the correlation between the foreground and the background, as well as the light and color effects of the entire image. A view can be assessed with more factors such as tune, contrast, space, focus, texture, framing, form, visual balance, perspective, etc. However, modeling a generalizable aesthetic standard can be a challenging task due to the difficulty of finding all the adaptive rule set. In our study, we primarily focus **compositional correctness** at best, while considering color, light, and content as weak control factors. We define such a good photo as AGV.

1.2 Mobile Camera Model

Camera pose can be decided by its position and orientation. Photo composition is a process to find the best camera pose that leads to the desired view. A simple camera model in CG conventionally has seven DoF - three in Cartesian coordinates to present camera position, three in Euler angles to present camera rotations, and one intrinsic parameter indicating the camera Field of View (FoV) (Christie et al., 2008). The camera model is shown in Figure 1.2. The camera in our research follows the model below.

The location of the camera in 3D space is represented with vector $\boldsymbol{p} = [x, y, z]$, where x, y, and z are the Cartesian coordinates. The orientation of the camera is represented with vector $\boldsymbol{o} = [\phi, \theta, \psi]$, where ϕ is the Euler angle of tilt (or pitch), θ is the Euler angle of pan (or yaw), and ψ is the Euler angle of roll. The FoV angle γ is a camera intrinsic parameter. Therefore, the camera model parameters are represented with vector $\boldsymbol{q} = [x, y, z, \phi, \theta, \psi, \gamma]$.

One advantage of using the above camera model is the scalability of applying it directly to a mobile camera. Particularly in our study, we consider the mobile camera to be a quadrotor drone mounted with a gimbal camera. The FoV (γ) is not considered to be a control parameter. Depending on the mechanical structure, a drone usually has seven or less DoF. Figure 1.3 shows an example of a drone camera that has five DoF in our study, which are aircraft translation on its *Pitch Axis*, *Yaw Axis*, *Roll Axis*, aircraft rotation around its *Yaw Axis*, and gimbaled camera pitch around its *Cam Pitch Axis*. The control parameters for the drone camera can be presented with $\boldsymbol{q} = [x_a, y_a, z_a, \phi_c, \theta_a]$, where the subscript a stands for aircraft, and the subscript c stands for the gimbal camera.

1.3 Problem Statement

We adopt the mobile camera model in Figure 1.2 and the drone model in Figure 1.3 described in Section 1.2. Our goal is to reduce drone control difficulty

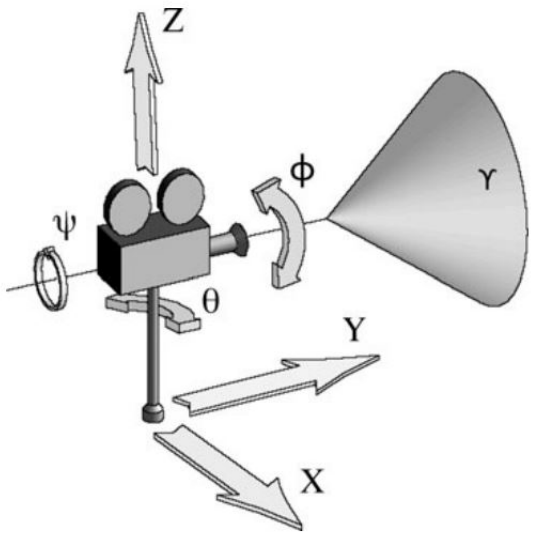

Figure 1.2.: A simple virtual camera model based on Euler angles: tilt (ϕ), pan (θ) and roll (ψ). Taken from (Christie et al., 2008).

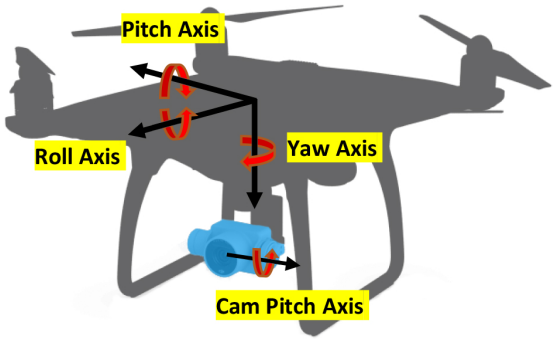

Figure 1.3.: A drone coordinate system. The drone camera has five DoF. Taken from (Kang et al., 2018).

during the composition process for capturing AGV. We break the problem down into two parts: (a) improve the drone control, and (b) simplify the composition process.

Figure 1.4.: The dual joystick RC (left) and gimbal camera (right) of a DJI© drone.

The manual control of a drone for photography tasks is sophisticated and requires training. The traditional way to fly a drone is by using a dual joystick RC (Figure 1.4 left). The asymmetric functions of the sticks make it challenging to conduct smooth eye-hand coordination. This situation is exacerbated with a gimbaled camera (Figure 1.4 right), which needs additional control. The *lost in space* pitfall is a significant challenge. Hanson and Wernert (1997) brought up the problem that is caused by managing high DoF camera in 3D environments. One given solution is to reduce the operational dimensions. The research XPose (Lan et al., 2017) shown in Figure 1.5 provides an intuitive touch-based interface, that can reduce the drone operation dimensions, for semi-autonomous photo shooting via points of view. However, they were not considering the gimbal operations, while gimbal plays an essential role in drone photography nowadays. Path customization has been intensively explored for drone photography or cinematography, such as trajectory planning by using pre-programmed command sets (Fleureau, Galvane, Tariolle, & Guillotel, 2016), key-frame positioning (Joubert, Roberts, Truong,

Berthouzoz, & Hanrahan, 2015; Roberts & Hanrahan, 2016), viewpoint optimization (Ngeli et al., 2017) shown in Figure 1.7. The focus of path customization methods is to design or optimize a trajectory that mostly depends on charted environments. However, the focus on HRI improvement is neglected while the drone is in mid-air for photography tasks.

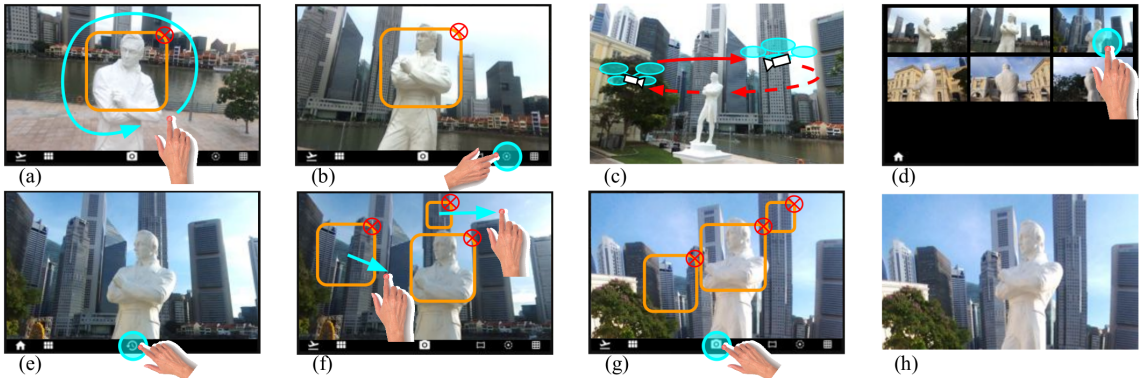

Figure 1.5.: An envisioned use case taken from (Lan et al., 2017). (a) Select an object of interest with the encircle gesture. (b) Activate the Orbit exploration mode. (c) The drone-mounted camera takes sample photos while orbiting the object of interest autonomously. (d) Browse sample photos in the gallery preview. (e) Restore a POV associated with a selected sample photo. (f) Compose a final shot by dragging selected objects of interest to desired locations in the photo. (g) Take the final shot. (h) The final photo.

The low level motor control of a drone distracts the user from the photography. A goal-oriented design is required to let the users forget about the drone and only focus on the high level tasks. An improved HRI solution is needed to make the drone controls more intuitive and natural.

On the other hand, as the views can be scored based on the photographic principals (Liu et al., 2010), the photo composition process can be further simplified, even automated, with a mobile camera. Although researchers have explored several autonomous solutions (Byers, Dixon, Goodier, Grimm, & Smart, 2003; Myung-Jin Kim et al., 2010; Zabarauskas & Cameron, 2014), previous approaches are limited to simple camera DoF and monotonous composition rules.

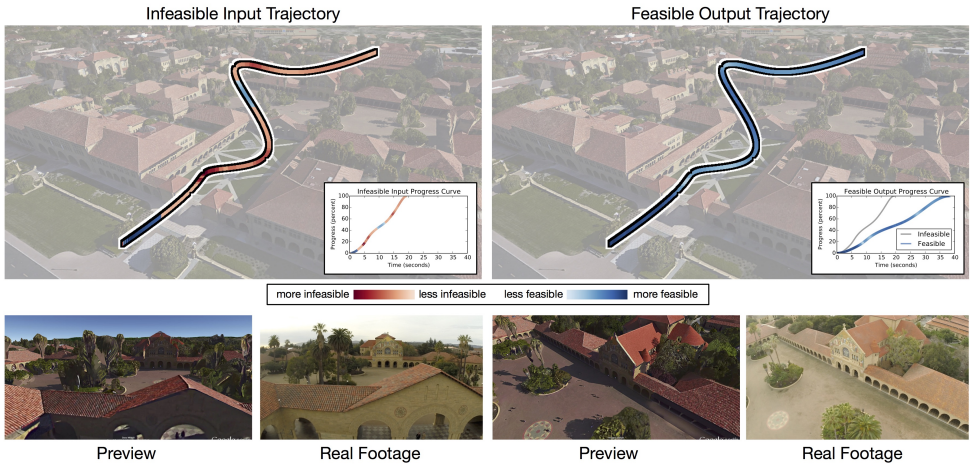

Figure 1.6.: The algorithm for generating feasible quadrotor camera trajectories. Taken from (Roberts & Hanrahan, 2016).

Figure 1.7.: Illustration of cinematographic framing constraints: size, viewing angle and position on screen (from left to right). Taken from (Ngeli et al., 2017).

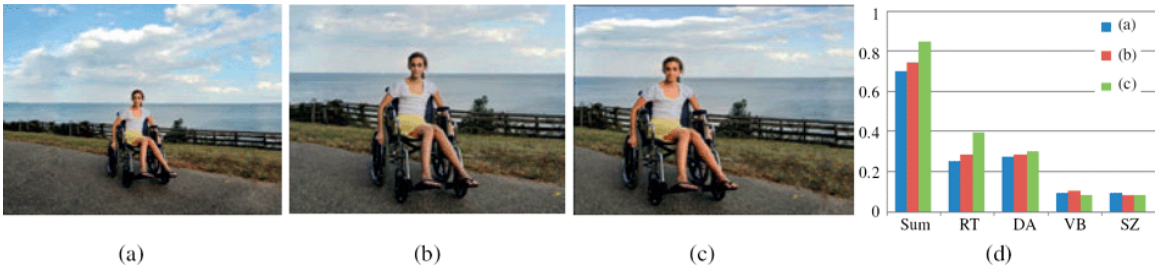

Figure 1.8.: Optimizing the aesthetics of the original photograph in (a) the method by Liu et al. (2010) leads to the new image composition shown in (c). (b) shows the cropping result of the baseline. The aesthetic scores are shown in (d). Our result in (c) obtains higher aesthetic score than (a). RT (rule of thirds), DA (diagonal), VB (visual balance), and SZ (region size) are components of the objective function.

The previous rule-based methods are not flexible enough to take photos with arbitrary targets in a complex scene. Recent breakthroughs in DL have enabled the automation in many complex tasks (He et al., 2016; Mnih et al., 2015; Silver et al., 2016). For example, RL framework allows agents to interact with the environment by taking actions based on sensory observations (Sutton & Barto, 2018). In our case, the drone takes fly actions to adjust its pose to find an appealing composition for the region of interest basing on the input camera view. AI algorithms can summarize appropriate strategies from extensive data for view composing.

Composing a good camera view is usually tedious and time-consuming with a drone. Other than the potential HRI improvement, existing automation methods rely on the prior knowledge of the working scene or nave composition rules, which have significant limitations in real applications. An automated view composing solution on a drone that can seek AGVs in uncharted scenes is needed to fill in the gap.

1.4 Research Questions

The current mobile camera view composing, including drone photography, has complicated and cumbersome operation process, which is not satisfactory. Our research focus on the improvements, which makes the main research question be:

1. Can we improve the view composing process with a mobile camera?

Moreover, we consider to make the improvements on a drone as the presentation of the mobile camera from two aspects: (a) the HRI on the control for photography tasks, and (b) the simplification of aesthetic view seeking and composing. Therefore, followed by the main research question, we form two sub-questions considering HRI listed below:

1.1 Is it possible to make the drone HRI more intuitive and natural?

Regarding the photography task, we further develop question 1.1 to:

1.2 How to let the user focus on photography tasks without thinking about drone motor controls?

On the other hand, with the consideration of the view composition topic, we propose question 2.1 as following:

2.1 Is it possible to simplify the drone photography process with aesthetic composition consideration?

And basing on the previous rule-based methods (Joubert et al., 2015; Zabarauskas & Cameron, 2014), we further develop the last question to be:

2.2 Is it possible to fully automate drone photography for uncharted scenes?

1.5 Scope

The scope of this research is divided into two sub-sections considering the HRI and automation respectively. However, both sections are aiming at improving the mobile camera view composition process. The sub-section on drone HRI improvement for photography is introduced in 1.5.1 and the drone photography automation sub-section is demonstrated in 1.5.2.

1.5.1 Multi-touch Gesture Controlled Drone Gimbal Photography

We first focus on the exploration of HRI improvements for drone photography described in Chapter 2. The conventional drone control relies on a dual joystick RC. However, the four DoF aircraft movements together with the gimbal camera pitch are mapping asymmetrically to the two joysticks, which leads to a long learning curve and are not intuitive. Even skillful users need to think about the aircraft controls that distracts photography tasks. A goal-oriented high-level HRI is needed for drone photography. We propose a novel framework for gimbal drone camera photography. Our approach unifies the drone aircraft with the gimbal as a single flying camera. The enhanced HRI allows the user to focus on the photography tasks instead of the motor control by applying simple touch gestures to

adjusting the flying camera on a mobile device. The gestures happen on the drone camera streaming as if the user is directly manipulating with the view. Our algorithm behind the scene seamlessly handles the gestures and interprets them into commands that combine the gimbal motion with the aircraft movement. Moreover, we utilize a sigmoidal gesture mapping function that compensates for abrupt drone swinging when moving horizontally, which ensures photo quality.

1.5.2 Region-of-Interest Based Reinforced Drone Photography

We further investigate the full automation of drone photography with aesthetic composition consideration (Chapter 3). Photographic principals such as *Rule-of-thirds* can be programmatic evaluated (Liu et al., 2010; Wei et al., 2018) and used to guide view composing (Zabarauskas & Cameron, 2014). However, most previous work relying on the heuristic rules do not scale well to arbitrary targets in complex scenes that require an overall aesthetic consideration of the foreground and background together with light, color, etc. Moreover, the rule-based visual servoing methods of mobile camera control are usually limited to low DoF (Byers et al., 2003; Myung-Jin Kim et al., 2010). Recent advances in RL allow us to handle complex control of agents with high performance (Gu, Holly, Lillicrap, & Levine, 2017; Lu et al., 2018). The photography procedure can be naturally integrated into the observation-action-reward RL settings. Therefore, we explore the reinforced approach on photography automation with a high DoF drone. We propose a framework that enables the drone to actively seek and compose visually appealing views in uncharted scenes basing on auto-detected or user-defined RoIs. The neural network requires training, and we utilize a photo-realistic 3D virtual environment for it.

1.6 Significance

The control of a high DoF mobile camera is complicated. Moreover, composing a well-framed photo with such camera requires knowledge and experience. The simplification of the view composing process with a mobile camera is needed. Existing methods rely on the prior knowledge of the working scene or simple composition rules, which have significant limitations in real applications. In this study, we particularly focus on a quadrotor drone equipped with a gimbal camera and improve its view composing by proposing a new drone HRI photography framework and a reinforced drone photography automation framework.

We claim our first contribution as a novel touch gesture based drone HRI that unifies the control of the movement of the aircraft and the gimbal. Our well-designed gestures enable the direct manipulation of the camera view instead of drone and gimbal operations. Such improvement significantly reduces the difficulty of drone photo composition process. User studies with 20 subjects show that our new HRI on the drone photography tasks leads to significant lower workload and better efficiency compared to the traditional RC control. The new interaction method has a high performance in both intuitiveness and easiness of navigation, which can be potentially applied to other teleoperation tasks in robotic control problems.

We claim our second contribution as a scalable RL solution for good view seeking and composing, that can be deployed to a drone for photography automation. The high DoF drone agent can actively explore the uncharted scene for framing RoIs with aesthetic consideration. We trained the agent under virtual 3D environments for cost-saving, efficiency, and safety. Our novel intermediate visual observations bridge the gap between virtual and real scenes and allow the drone to execute autonomous framing in practical applications. The real photography task results show that our method outperforms heuristic baseline method suggested by Zabarauskas and Cameron (2014), and produce visually appealing drone photos

as good as human operators. David Silver (Silver, 2016) asserts that: $Artificial\ Intelligence = Reinforcement\ Learning + Deep\ Learning$. Our study makes necessary exploration and accumulation for general AI research. Our RL solution has the potential to be used in other drone automation tasks, such as scanning, by redefining the rewards.

1.7 Summary

This chapter provided the background of the view composing problem with a mobile camera. We introduced the mobile camera model that this study uses. We further discussed the research problem in two-fold, considering the improvement of drone HRI and automation of photography task. In the next two chapters, we are going to cover two aspects of the problem respectively in details.

CHAPTER 2. FLYCAM: MULTI-TOUCH GESTURE CONTROLLED DRONE GIMBAL PHOTOGRAPHY

This chapter has been published in IEEE Robotics and Automation Letters and presented at IROS 2018 (Hao Kang, Haoxiang Li, Jianming Zhang, Xin Lu, and Bedrich Benes, 2018). The first authors contributions include implementation, testing, and manuscript preparation. Coauthors provided ideas, resources, and guidance in the research and helped in proofreading and modification in the manuscript preparation.

2.1 Abstract

We introduce FlyCam - a novel framework for gimbal drone camera photography. Our approach abstracts the camera and the drone into a single flying camera object so that the user does not need to think about the drone movement and camera control as two separate actions. The camera is controlled from a single mobile device with six simple touch gestures such as rotate, move forward, yaw, pitch, etc. The gestures are implemented as seamless commands that combine the gimbal motion with the drone movement. Moreover, we add a sigmoidal motion response that compensates for abrupt drone swinging when moving horizontally. The smooth and simple camera movement has been evaluated by user study, where we asked 20 human subjects to mimic photograph taken from a certain location. The users used both the default two joystick control and our new touch commands. Our results show that the new interaction performed better in both intuitiveness and easiness of navigation. The users spent less time on task and the System Usability Scale index of our FlyCam method was 75.13 which is higher than the traditional

dual joystick method that scored at 67.38. Moreover, the NASA task load index also shown that our method had lower workload than the traditional method.

2.2 Introduction

The consumer civilian drone technology has became increasingly accessible and affordable. Many advances have been dedicated towards longer flight time, collision avoidance and path customization. Consumer drones are also often equipped with a high-quality camera mounted on a rotatable gimbal that is controlled separately. People most commonly fly the drones to obtain impressive videos or to take pictures. In a typical configuration, the real-time drone camera streaming is viewed with the help of a mobile application running on a smart phone or a tablet. Some drones store the videos on the on-board memory card that can be viewed later.

Most of the technological progress has been dedicated to the drone themselves and the most common way to control them is by using a dual joystick remote controller (RC), where one joystick is used for turning the drone and the other joystick is for propelling. The gimbaled camera needs additional control that complicates the navigation. While commonly used in amateur and professional planes and drones, this kind of navigation is not intuitive for beginners. The dual joystick operation asymmetry leads to a long learning curve for the starters and has caused many failures and destroyed drones. Even skillful users need to take into account additional consideration for drone control that is distracting when a particular objective, such as a photo or a video, is being targeted. This situation is exacerbated with drones with a separate camera control. In order to get a desired view, the user must steer the drone to reach an approximate location, then adjust camera orientation to see the resulting view. If the view is not as expected, the drone needs to be moved further, camera adjusted, etc. The user usually needs to iterate this process to achieve the desired camera view.

Our key observation is that the Human-Drone Interaction could be more intuitive and natural if one would decouple the mechanical control from the desired objective. A goal-oriented design would let the users forget about the drone and only focus on the high level tasks. The low level motor control would be abstracted out from the users and the users should be able to operate the views with their flying camera directly, rather than worrying about the direction where the drone has to go. Moreover, as a derived camera application running on mobile devices, the drone photography applications could naturally integrate common touch gestures for camera and drone controls to replace the RC.

In this paper, we introduce FlyCam, a multi-touch camera view manipulation framework for drones equipped with cameras with gimbal. Our framework substitutes the traditional RC for drone controls by simplifying the low level aircraft controls, together with gimbal operations, to only six simple and intuitive multi-touch gestures. A single finger drag rotates the aircraft and camera; a double finger drag drives the drone up/down or left/right; and a single/double tap hold moves the drone forward or backward along the camera optical axis. The speed of the drone actions are controlled by the dragging distance on the screen or the tapping pressure. The direct manipulation of the camera view instead of drone and gimbal operations significantly reduces the difficulty of photo composition process with drone camera. The difference is reflected clearly in Figure 2.1 as trajectories extracted from one user study of five shooting tasks with a drone (2.7.3).

We have performed a user study, where we compare the traditional RC control with our new interface. The results show that our framework offers better efficiency in the drone photography tasks, and our new interface provides a better usability and a lower workload to the users. Our main contribution is in providing a unified framework that encapsulates control of movement of drones and camera control.

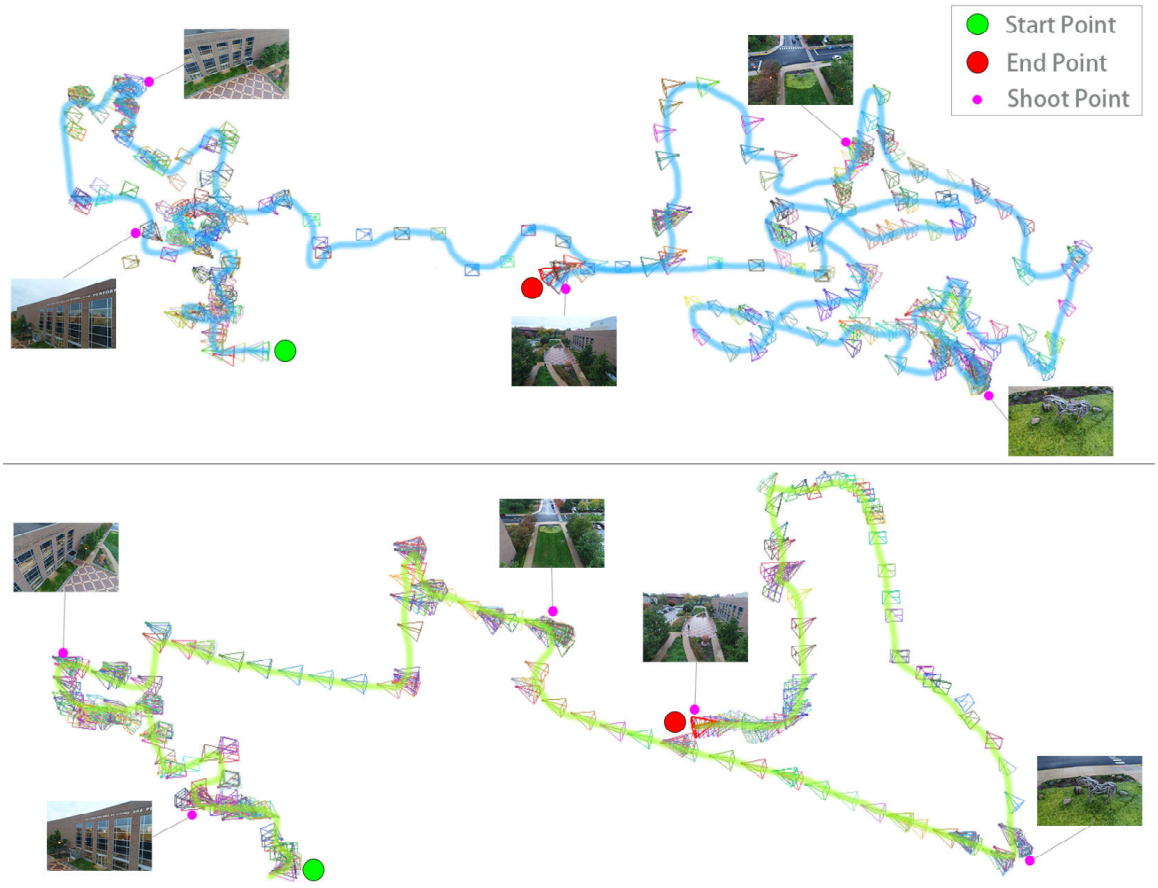

Figure 2.1.: Comparison of the drone trajectories taken by the traditional dual joystick RC method (top) and our new FlyCam method (bottom). The goal of the experiment was to recover five views given as photographs. The top trajectory is longer and more intricate, indicating that the user had to perform more adjustments and put more efforts during the task. The bottom trajectory is more direct and concise, indicating that the user was able to get to the desired locations quickly and fine-tune the position better by using our method.

2.3 Related Work

We relate our work to the control of gimbaled camera and aircraft and to the touch gestures.

2.3.1 Gimbaled camera and aircraft control

The gimbaled camera control of UAVs was discussed before the consumer drones becomes popular by Quigley, Goodrich, Griffiths, Eldredge, and Beard (2005) and Jakobsen and Johnson (n.d.). Two studies of fly-by-cameras (DongBin Lee, Vilas Chitrakaran, Burg, Dawson, & Bin Xian, 2007; Neff, Lee, Chitrakaran, Dawson, & Burg, 2007) analyzed the kinematics models of drones and camera and motivated our work. Drone manufactures have also introduced various First-Person View (FPV) displays (DJI, 2017a; Yuneec, 2016a). The displays can track the head pose of the user, and reflect action to the drone gimbal. Contrary to the previous work, the novelty of our work is that the user can control both the gimbal and the drone movement by touch gestures from a single mobile device.

The most conventional method of consumer drone controls relies on a dual joystick Remote Controller (RC) and it is commonly used for example in works (3D-Robotics, 2017; DJI, 2017d; Yuneec, 2016a). For lighter, smaller, and more affordable consumer drones, the RC is replaced by a mobile application that uses on-screen virtual joysticks (Parrot, 2012) or device built-in accelerometers to control the drone (Ehang, 2016). However, this requires the users to have an understanding of drone dynamic behavior, which are not designed naturally for efficient and undemanding photo composition. This often causes navigation errors and even damage to the UAVs.

Prior research on human robot interactions proposes a number of novel drone controls. Various hands free control methods, such as eye tracking (Ettikkalayil, 2013; Hansen, Alapetite, MacKenzie, & Møllenbach, 2014), speech (Landau & van Delden, 2017; Trujillo, Puig-Navarro, Mehdi, & McQuarry, 2017), and brain electroencephalogram (LaFleur et al., 2013; Y. Yu et al., 2012), were applied to control UAVs. Body gestures were also widely studied and some rely on external sensors to capture the gestures, such as Microsoft Kinect (Ng & Sharlin, 2011; Pfeil, Koh, & LaViola, 2013; Sanna, Lamberti, Paravati, Henao Ramirez, &

Manuri, 2012), the Leap Motion controller (Chandarana, Trujillo, Shimada, & Danette Allen, 2017; Sarkar, Patel, Ram, & Capoor, 2016), or wearable devices (Sandru et al., 2016; Teixeira, Ferreira, Santos, & Teichrieb, 2014). Other methods use the on-board cameras or sensors to guide a single UAV or a team of UAVs (Lichtenstern, Frassl, Perun, & Angermann, 2012; Monajjemi, Wawerla, Vaughan, & Mori, 2013; Nagi, Giusti, Caro, & Gambardella, 2014; Nagi, Giusti, Gambardella, & Di Caro, 2014). Empirical studies on Human-Drone Interaction (HDI) using body gestures were conducted to explore the natural human behaviors in the interaction scenarios (Abtahi, Zhao, E., & Landay, 2017; Cauchard, E, Zhai, & Landay, 2015; E, E, Landay, & Cauchard, 2017; Obaid, Kistler, Kasparavičiūtė, Yantaç, & Fjeld, 2016). Multi-modal UAV controls were also used to gain better control over hybrid modes. The combinations of speech, gesture (hand and body), and visual markers were applied by Peshkova, Hitz, and Ahlström (2017) and Fernndez et al. (2016), and they were compared and discussed by Abioye et al. (Opeyemi, Prior, T Thomas, Saddington, & Ramchurn, 2019). The nontraditional input modalities were analyzed to form a scheme in developing intuitive input vocabulary (Peshkova, Hitz, & Kaufmann, 2017). However, it is difficult to translate natural vocabulary into drone instructions for precise control.

Path customization was explored as a task level UAV control and some results have been successfully applied to the consumer drone industry. The path customization is mainly set up for drone photography or video recording trajectory planning by using pre-programmed command sets (Fleureau et al., 2016), key-frame positioning (Gebhardt, Hepp, Nägeli, Stevšić, & Hilliges, 2016; Joubert et al., 2015; Roberts & Hanrahan, 2016), viewpoint optimization (Nägeli, Meier, Domahidi, Alonso-Mora, & Hilliges, 2017; Ngeli et al., 2017), way-point setting, and following the user motion (3D-Robotics, 2017; DJI, 2017d; Yuneec, 2016b). Existing systems enable designing cinematography shots ahead of time in a virtual environment. In contrast, our system makes it easier to perform artistic exploration

while the drone is in mid-air, which could be useful, e.g., to explore how the scene looks in real-world lighting conditions

2.3.2 Touch gestures

Researchers explored and defined natural multi-touch gestures with 3D objects on large screens (Buchanan, Floyd, Holderness, & LaViola, 2013; KUa & Chen, 2014), as well as single-touch techniques for virtual camera manipulation on small devices (Fiorella, Sanna, & Lamberti, 2010; Mendes, Sousa, Ferreira, & Jorge, 2014). Navigation in virtual 3D environment using multi-touch gestures were also investigated (Jankowski, Hulin, & Hachet, 2014; Ortega, 2014) and these studies are instructive for multi-touch gesture design for drone navigation, but they were focusing on virtual 3D environment.

Multi-touch gestures have been applied in UAV Ground Control Station (GCS) (Crescenzio, Miranda, Persiani, & Bombardi, 2009; Haber, 2015; Haber & Chung, 2016), and experimented in Human-Robot Interaction (HRI) in the context of teleoperations (Paravati, Sanna, Lamberti, & Celozzi, 2011), bipedal walk (Sugiura et al., 2009), and general control (Micire, Drury, Keyes, & Yanco, 2009). Close to our work is the research of Chen et al. (Chen, Lee, Chan, Liang, & Chen, 2015) and Gross (Gross, 2016) who introduced methods to operate a drone through camera view manipulation with multi-touch gestures. A more recent research XPose Lan et al. (2017) also provides an intuitive touch-based interface for semi-autonomous photo shooting via points of view. The main difference to our work is that the other studies were not considering the gimbal operations, while gimbal plays an important role in drone photography nowadays. Contrary to our work, the gestures are used to navigate the drone movement and not to unify movement with the control of the camera.

2.4 System Overview

FlyCam framework consists of four modules shown in Figure 2.2. The application runs on a mobile device, takes as input multi-touch gestures, and visualizes the drone camera streaming as the output (see also Figure 2.7 for the graphical user interface and the accompanying video for real-time demo).

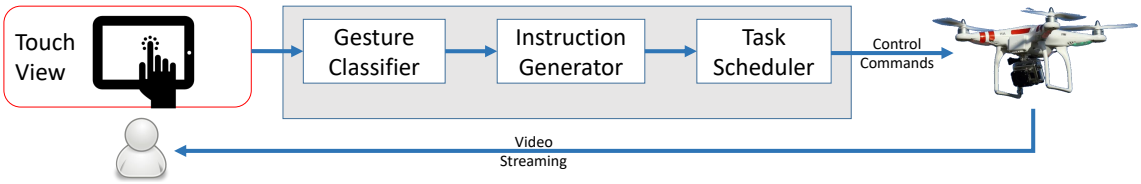

Figure 2.2.: Overview of the pipeline for FlyCam framework. The user inputs gestures that are classified and instructions for the drone navigation are generated and scheduled. Visual feedback is immediately shown on the screen.

The *Touch View* that takes multi-touch gestures as the input. The *Gesture Classifier* detects and categorizes the user input into meaningful gestures and parameterizes them. For example, moving one finger to the left is interpreted as rotating left. The distance of the stroke is calculated as the parameter of the corresponding angle. The *Instruction Generator* converts the gestures and their parameters into drone control instructions that are sent to the drone as a commands. This block also unifies the heterogeneous operations between the gimbal and drone. Finally the *Task Scheduler* communicates directly with the drone.

2.5 Gesture Control

Because of the landscape orientation of the streaming video from the drone, the mobile device is held horizontally. Users prefer to use two thumbs to perform touch gestures on small screen devices and they hold the device with one hand, and perform touch gestures with the other hand alone on large screens as shown in Figure 2.3.

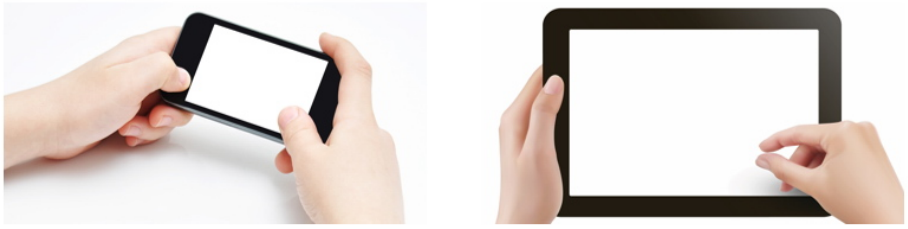

Figure 2.3.: Holding behavior of the control tablet can vary for different sizes of the screen. Small screens are controlled with thumbs, whereas large screens are controlled by one hand.

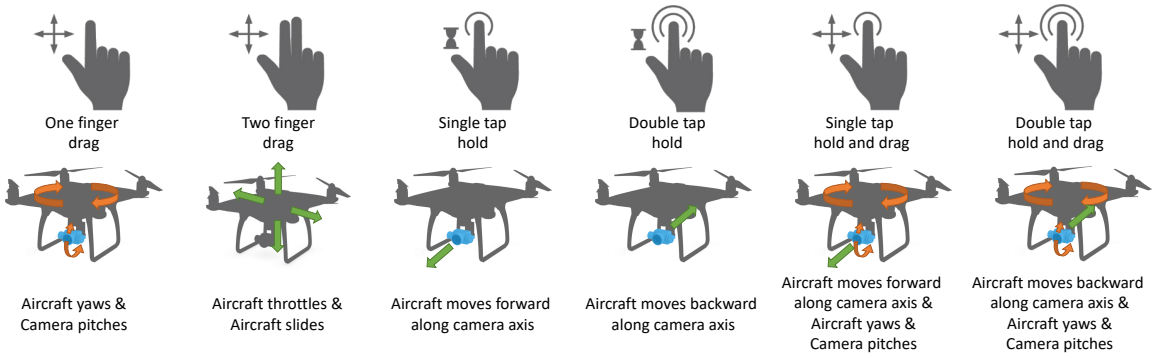

Figure 2.4.: Our six gestures and the corresponding drone actions.

We employed four atomic *gestures* in FlyCam framework that are combined into six gestures that serve well for both holding behaviors from Figure 2.3. The atomic gestures are one or two finger drag, single tap hold, and double tap hold. These four gestures can be performed easily with two thumbs, as well as one single hand.

Figure 2.4 and the accompanying video show the six gestures used in FlyCam framework and the corresponding mapping to the drone actions. These six touch gestures constitute the user input that is captured, parsed, and abstracted by the four modules of the framework. The framework allows to fly the camera freely without the user needing to concentrate on the low level drone controls. It also seamlessly links the aircraft movement with the gimbaled camera operation, which provides a more user-friendly fly-by-camera mode (see Section 2.6.3).

2.6 System

The six gestures from Section 2.5 are implemented in our system that can recognize them from the touch screen, interpret, and send as control commands to the actual drone (see Figure 2.2).

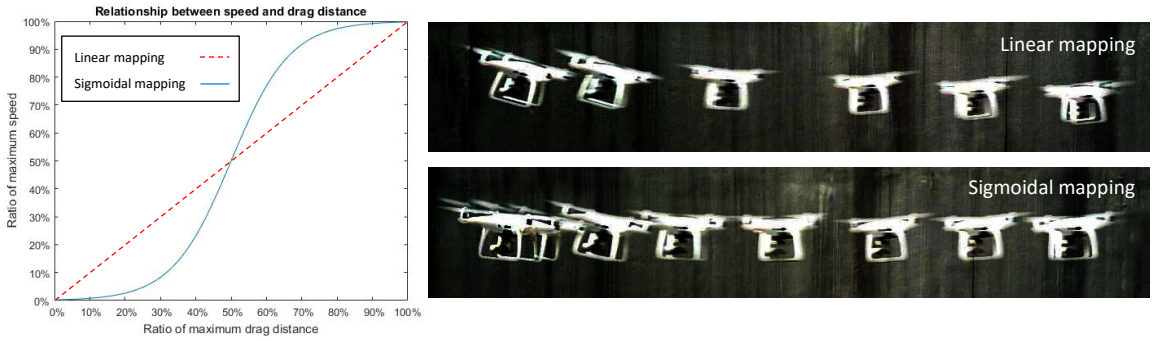

Figure 2.5.: Linear mapping of the velocity and the drag distance cause the drone to move abruptly and sway at the beginning and at the end of each gesture (top right trajectory). We compensate for this behavior by using a sigmoidal function, and the corresponding trajectory is shown on the bottom right.

2.6.1 Touch View

Touch View is the interaction layer of the framework. It provides the Graphical User Interface (GUI) (shown in Figure 2.7) and takes the multi-touch operations as user input. Touch View also receives, decodes, and presents drone camera streaming in real time that allows the user to see immediate visual feedback of their operations.

2.6.2 Gesture Classifier

The *Gesture Classifier* identifies touch gestures and categorizes them into the types of drone action by converting them into parameterized actions for generating

drone instructions. The conversion parameterizes the drag distance in the screen $x - y$ coordinate and the touch pressure for tap hold action.

2.6.3 Instruction Generator

We assume a drone with five Degrees of Freedom (DoF) and the associated coordinate is shown in Figure 1.3. The five DoF are: 1) translation on roll axis, 2) translation on pitch axis, 3) translation on yaw axis, 4) rotation around yaw axis, and 5) rotation around camera pitch axis.

The movements of the aircraft and the gimbaled camera are controlled by a combination of the velocities on the five DoF - three line velocities and two angular velocities. The parameters received from the *Gesture Classifier* contains drag distance or touch pressure, together with drone action type - translation on camera optical axis, slide, throttle, yaw, and gimbaled camera pitch. The drag distance and touch pressure are used for determining the speeds. For the action of translation on camera optical axis, the stronger the pressure is, the higher the speed is. The pressure is retrieved from the device as a float pointing number in range $[0.0, 1.0]$. For the slide, throttle, yaw and gimbaled camera pitch actions, the larger the drag distance is, the higher the speed is. The relationship between the speed and drag distance is shown in Figure 2.5 and we use two kinds of mapping:

$$r_v = c||p_1 - p_0|| \tag{2.1}$$

$$r_v = 1/\left(\mathrm{e}^{-12||p_1-p_0||+6} + 1\right), \tag{2.2}$$

where r_v is the ratio of drone maximum velocity, and p_0 and p_1 are gesture start and end points, and c (Eqn (2.1)) is a scalar constant depending on device resolution. The mappings are shown and compared in Figure 2.5. The simple linear mapping in Eqn (2.1) causes the drone to accelerate fast and overshoot at the end. We experimentally observed that the logistic function mapping (Eqn 2.2), which is used in our implementation, compensates for the weight of the drone and provides

smoother and more coherent drone trajectory which leads to stable images and better user experience. The second row of Figure 2.5 shows that the trajectory is nearly as horizontal as directed.

Our main contribution is the union of the heterogeneous operations between gimbal and aircraft that is achieved by redefining the forwards and backwards actions. Rather than being relative to the drone heading direction, these two actions are changed to be relative to the camera optical axis. The horizontal speed (on roll axis) and vertical speed (on yaw axis) of the drone can be calculated with orthogonal decomposition on forwards or backwards speed:

$$\begin{bmatrix} v_h \\ v_v \end{bmatrix} = \begin{bmatrix} 0 & \cos\alpha \\ \sin\alpha & 0 \end{bmatrix} \begin{bmatrix} v_f \\ v_f \end{bmatrix} \tag{2.3}$$

$$v_f = r_v v_{max} \tag{2.4}$$

where v_h is the horizontal and v_v the vertical component of the forward velocity v_f, and α is the camera pitch angle relative to the horizontal plane. The forward velocity v_f is a portion of the drone maximum velocity v_{max} determined by the ratio r_v from Eqn 2.2.

Figure 2.6 shows the velocity decomposition and the trajectory comparison of the two methods on a diagonal motion towards a target. The trajectories reflect the operation simplification brought by FlyCam method.

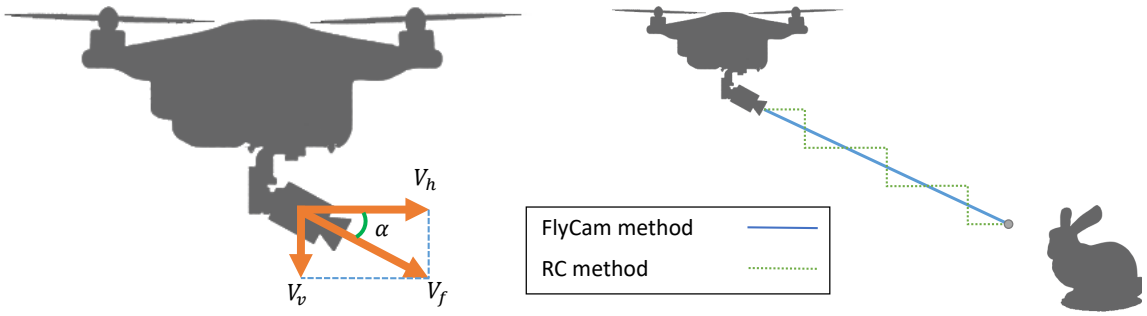

Figure 2.6.: Our unified control of the drone motion and camera motion allows for a smooth transition between the motion and camera aiming.

The drone command set is constructed with the velocity information that is based on the ratio of max speed for each DoF and the velocity decomposition.

2.6.4 Task Scheduler

The communication from the framework to the drone is executed by the *Task Scheduler* module. This module maintains a thread that periodically reads the afore-described velocities inside the command set [v_{pitch_axis}, v_{roll_axis}, v_{yaw_axis}, ω_{yaw}, ω_{cam_pitch}] from the *Instruction Generator* module and sends instructions to the gimbal and aircraft respectively.

2.7 Implementation and Evaluation

2.7.1 Implementation

We have developed the application and we tested it by using DJI Phantom 4 Pro (DJI, 2017d). Our framework was implemented in Java on a 9.7" Android tablet (ASUS ZenPad 3S 10) and a 5.2" Android phone (Huawei P9). We have used DJI Mobile SDK for Android 4.3.2 (DJI, 2017b) and DJI UILibrary for Android 1.0 (DJI, 2017c).

Figure 2.7 shows the GUI of FlyCam framework. The GUI is displayed on the top of the real-time camera streaming. The top status bar (#1) indicates the information such as the pre-flight aircraft status, GPS and remote controller signal strength, remaining battery power, etc., #2 indicates two buttons for drone taking off/landing and gesture mode activating/deactivating, #3 indicates the camera widget for photo shooting and video recording as well as advanced settings. The dash board widget (#4) provides the aircraft compass, as well as some in-flight information such as distance, altitude, and velocity. The traces in the center of the screen (#5) are examples of multi-touch gesture that have been applied to the

framework, in the case of Figure 2.7 double finger drag: aircraft throttle up is displayed.

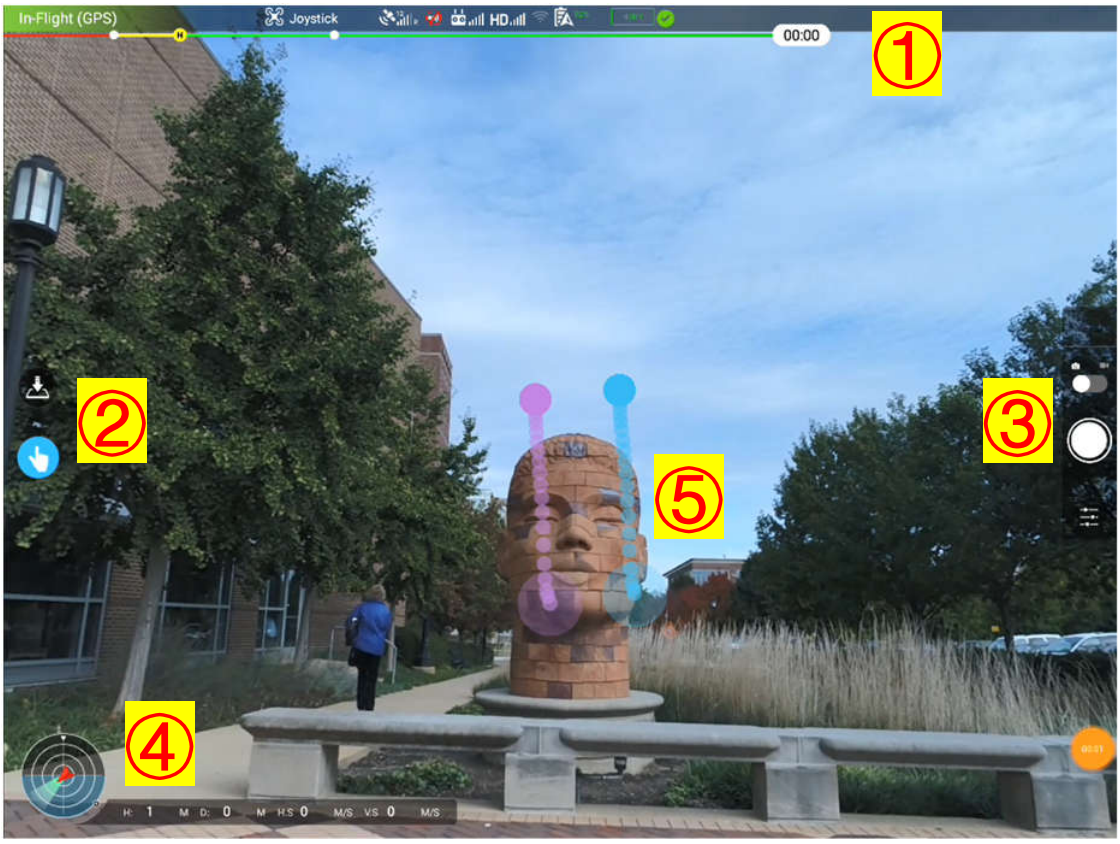

Figure 2.7.: The Graphical User Interface of FlyCam framework.

2.7.2 System Evaluation

The application provides real-time feedback and the timing of the individual system modules from Figure 2.2 is shown in Table 2.1.

One touch gesture can be classified and turned into corresponding drone commands within a few milliseconds. The task scheduler module executes a command every 20 milliseconds to load and send out commands. The bottleneck of the framework implementation is the communication between the framework and

Table 2.1.: Module timing in [ms].

Gesture Classifier	0.18
Instruction Generator	3.79
Task Scheduler	20.32
Framework & Drone Communication [a]	33.60
Sum	57.89

[a]The latency was measured with a distance of 30 meters in the open space with a strong signal.

the drone which is a limitation of the hardware and the underlying SDK. The speed of our application is sufficient to provide complete control over the drone.

2.7.3 User Study

We conducted a comparative user study between the traditional RC and FlyCam method. All participants were exposed to both approaches and they were also asked to capture the same set of photographs. A post scenario survey was made by using the System Usability Scale (SUS) and The NASA Task Load Index (NASA-TLX). The results were compared and analyzed for four criteria: 1) photo similarities, 2) task time spent, 3) SUS score (Brooke, 1996), and 4) NASA-TLX score (Hart, 2006; Hart & Staveland, 1988). These measurements evaluate how quickly and how easily the participants were able to get a desired photograph (Figure 2.8).

Participants

Our user study included 20 volunteers (50% female and 50% male) of ages 19–33 years with the mean of $\mu = 23$. The participants have background in technology (12), engineering (3), design (3), science (1), and management (1). None of the participants had any prior drone operation or related experience.

Apparatus, Setting, and Tasks

The study was conducted outdoors with the drone Obstacle Avoidance (OA) sensors fully activated. The participants were supervised by a certified professional drone operator (guide) for the whole study for safety consideration.

We prepared five photos (ground truth) that were taken in advance on the test site (the photographs as well as the photos taken by the users are available as additional material). The ground truth photographs include significant visual point taken from varying angles, ranges, and compositions. The tasks are to reproduce the given ground-truth photos. The sequence of the ground truth photos was fixed. Without setting any time limit, each study took about $45 - 60$ minutes including demonstration time, drone testing time, talk time, exit survey time, etc.

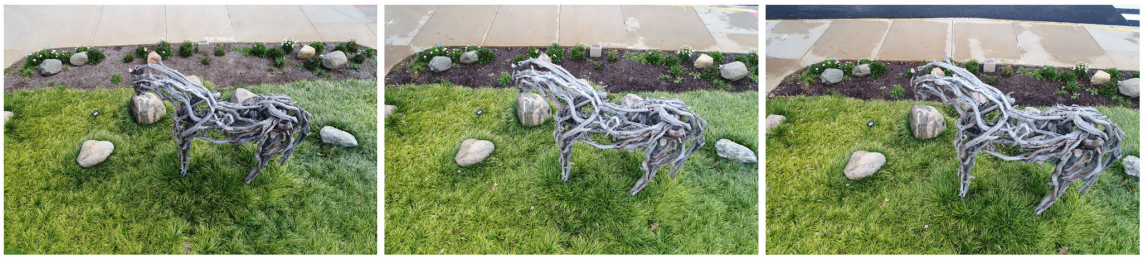

Figure 2.8.: The ground truth photo (left), a photo taken with FlyCam method (middle), and with the traditional RC method (right).

Procedure

For each participant, we randomly decided the order of the two methods to avoid the sequential effect of tested methods as suggested by A. J. Yu and Cohen (2008).

After the testing order had been decided, a brief introduction of the drone and the tested method was given to the participant. The participant then had three minutes for a test flight in order to get familiar with each method of control. Before the actual testing, the five tasks were introduced and explained to the participants.

The drone was started by the certified guide, recording was turned on, and then the control device was passed to the participant. The participant was shown the hard copies of the ground truth photos one by one and was asked to reproduce the photos. The average time to complete the tasks for traditional RC method was 7 minutes 34 seconds, and for FlyCam method was 7 minutes 02 seconds. During the tasks, the participant was allowed to ask the guide about the usage if it was needed. When the participant finished the last task, the screen recording was stopped and the drone was landed by the guide.

After both the methods were tested, the participant was asked to complete a web-based exit survey. The survey as well as the testing were anonymous, and included demographic information, SUS questionnaire, and NASA-TLX assessment. The survey took $10 - 15$ minutes to complete. Moreover, we have also recorded the time spent for each photograph. The screen recording video, and the photos taken by the participants were archived.

2.7.4 Results

Similarity of photo composition

We contrasted the photos taken by the participants with the ground truth photos. In order to compare the photograph compositions, we calculated the camera position and orientation (quaternion form) when each photo was taken. This information was recovered with the help of VisualSFM (Wu, 2011; Wu, Agarwal, Curless, & Seitz, 2011) for each photo. We computed the camera position change (Δt) and rotation change (Δr) between each user taken photo and the corresponding ground truth photo. These changes were categorized by methods, and tested with two Matched Pairs t Tests respectively on the population mean differences of Δt and Δr. The results show that the data do not provide evidence of significant differences for the two methods on either Δt or Δr

($\mu_{diff\_\Delta t} : DF = 99, t = 1.26, P - value = 0.2106, \alpha = 0.05; \mu_{diff\_\Delta r} : DF = 99, t = 0.78, P - value = 0.4373, \alpha = 0.05$).

Considering the outdoor environment, the drone position and camera orientation were heavily affected during the tasks by the external conditions such as wind, which created randomness to a certain extent. As figure 2.8 shows, the participants were using the same standard to recover the photos with the two tested methods and we did not expect and observe similarity difference in photo composition in the study.

Timing

Whereas both methods can achieve the same result, an important measure of the suitability of each method it the actual time spent in achieving this goal. The mean time spending of the 20 participants by using FlyCam method is 422.35 second with a standard deviation of 88.23. The mean time spending of the 20 participants using traditional RC method is 453.95 second with a standard deviation of 111.16.

A Matched Pairs t-Test on the population mean difference of time spent between the two tested methods shows the data provide evidence that there is a significant difference between the time spent on task completion using the two methods ($DF = 19, t = -2.10, P - value = 0.0496, \alpha = 0.05$). Based on this, we conclude that FlyCam method shows a better efficiency than the traditional RC method in photo composition tasks. This can be attributed to the fact that FlyCam method combines the aircraft motion and camera operation effectively, which makes the drone reach the target zone more quickly. Also, FlyCam method makes the fine tuning process easier and saves a lot of unnecessary camera pose adjustments. The shooting positions of the five ground truth photos in the experiment are relatively independent to each other. FlyCam method can work more efficiently in continuous scenes for view selections.

System Usability Scale

The System Usability Scale (SUS) (Brooke, 1996) is widely applied reliable tool for measuring the usability. The SUS consists of 10 item on 5 Likert scale response (strongly disagree, disagree, neutral, agree, and strongly agree) questionnaire in our post scenario survey for both tested methods. The questions we asked were:

1. I think that I would like to use this method frequently.
2. I found this method unnecessarily complex.
3. I thought this method was easy to use.
4. I think that I needed or would need help to recall the usage of this method.
5. I found the various human-drone interactions in this method were well integrated.
6. I thought there was too much inconsistency (unexpected drone poses/behaviors) in this method.
7. I would imagine that most people would learn to use this method very quickly.
8. I found this method very cumbersome to use.
9. I felt very confident using this method.
10. I needed to learn a lot of things before I could get going with this method.

We applied the scoring system suggested by Brooke et al. (Brooke, 1996). The mean score of FlyCam method was 75.13, which is higher than the overall score of the traditional RC method that was 67.38. A research on 3500 SUS surveys within 273 studies (Bangor, Kortum, & Miller, 2009) gave out a total mean score of 69.5, which shows that FlyCam framework is above average and therefore better than the traditional RC method from the system usability perspective.

From the responses of question 7 and question 10, 47.5% of the participants highly agreed that FlyCam method can be learned quickly and easily, while only 35% thought so for the traditional RC method. This reflects that the learning curve of FlyCam method is less steep than the RC method to more users. Moreover, once the user was comfortable with the operations, FlyCam method gains more fidelity. After getting familiar with the methods and completing the tasks, 85% of the participants preferred to use FlyCam method frequently basing on question 1 response.

The NASA Task Load Index

Besides system usability, we also evaluated the user workload. The NASA Task Load Index (NASA-TLX) Hart (2006); Hart and Staveland (1988) is a subjective multidimensional assessment tool to rate the workload of tasks or system. Our post scenario survey includes NASA-TLX rating scales due to the essence of our research being UAV operations. The workload is detached into six factors in NASA-TLX, which are 1) Mental Demand (MD), 2) Physical Demand (PD), 3) Temporal Demand (TD), 4)Overall Performance (OP), 5) Effort (EF), and 6) Frustration (FR). The overall workload score of FlyCam method is around 36, which is four points less than 40, which is the workload score of the traditional RC method. The comparison of the calculated workload index average weighted rating scores is shown as Figure 2.9. It shows that except for the OP every other factors of FlyCam method have a lower rate than the traditional RC method. However, the difference in OP is not statistically significant:
$(DF = 19, t = -0.3996, P - value = 0.6939, \alpha = 0.05)$. With FlyCam method, user shifted more attention from PD to MD and OP. This phenomenon was also evidenced by a study on large multi-touch Ground Control Station of UAVs (Haber & Chung, 2016). The multi-touch gestures free the users from monotonous and repetitive physical operations and allows them to put more efforts in thinking and

getting better performance on photo composition. The NASA-TLX rating scores indicates that FlyCam method had lower workload to the participants than the traditional RC method for the drone photography tasks.

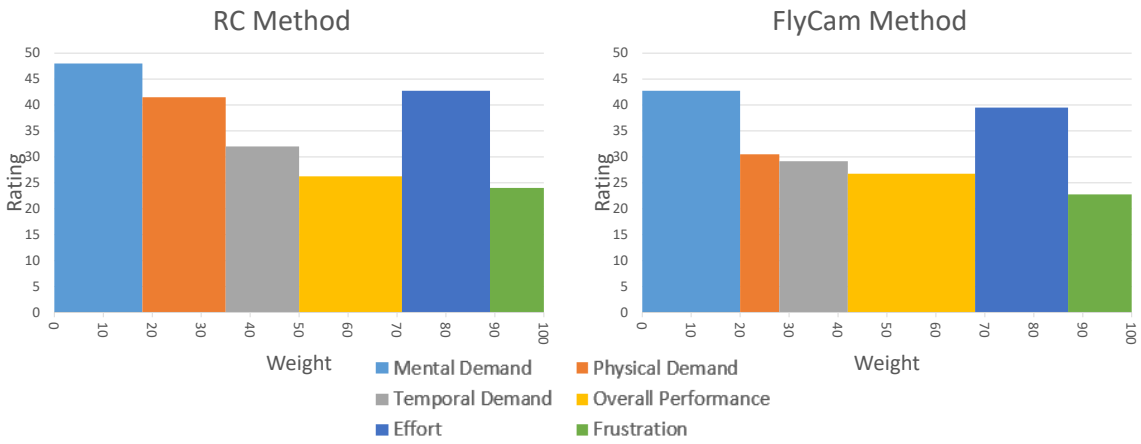

Figure 2.9.: Comparison of the calculated workload index average weighted rating scores. FlyCam shows lower workload in MD, PD, TD, EF, and FR, whereas the traditional RC method shows a slightly (0.5) lower workload in OP.

2.8 Conclusions

We introduced FlyCam, a novel framework that enables users to easily take photographs with drones equipped with gimbal. Our key contribution is in decoupling the flight from the camera operations. The user simply navigates the drone as if it was a flying camera capable of free movements in 3D space and FlyCam framework takes care of the drone and camera control. We introduced six simple touch gestures to utilize this unified control model. We further introduced several novel techniques such as mitigating the swaying of the drone by using sigmoidal velocity control and moving the gimbal in sync with the drone rotation.

We evaluated our system with a user study, where the users were asked to replicate given photographs. Our evaluation shows that FlyCam method outperforms the traditional two joystick control in terms of readiness of completion and easiness of usage. FlyCam method also scored higher in the NASA Task Load

Index (Hart, 2006; Hart & Staveland, 1988) as well as in System Usability Scale (Brooke, 1996).

Our system has several *limitations*. First, there is a communication delay caused by the hardware that causes lagging of the response. We assume this will be addressed by new drones and in a new version of the SDK. Second, the tap hold gesture is not accepted naturally by all users. Two of the users habitually applied double tap hold instead of when they actually intent to do a single tap hold, potentially due to the habit on mouse left button double-click. Besides, the delayed response of tap hold gestures makes the drone position adjustment in close range jerky. A potential replacement gesture could be a pinch, which can zoom in and out the view by driving the drone closer or further to the camera view center along the optical axis. Third, while we aimed at keeping the number of touch gestures minimal, it could be possible to extend the number of gestures as it is not obvious what a good small number of gestures would be.

There are several possible avenues for *future work*. The FlyCam has been tested only on one drone equipped with gimbal camera and it would be interesting to see how this approach can be generalized to different drones. Another future work would include comparison of the gestures on tablets of different sizes. We have observed in our user study that it is not always intuitive for the users to make the mental mapping of the screen size to the desired action of the drone. Another future work would be to include left-handed subjects. The gestures are symmetrical and it should be easy to consider.

2.9 Acknowledgment

This work was supported by Adobe Research. The authors would like to thank Suren Deepak Rajasekaran for the help on video editing, and Booker Smith for useful discussions. The authors also would like to thank the anonymous reviewers for valuable feedback.

CHAPTER 3. DR^{3}CAM: REGION-OF-INTEREST BASED REINFORCED DRONE PHOTOGRAPHY

This chapter is a result of collaboration with Yichen Sheng, Jianming Zhang, Suren Deepak Rajasekaran, Zhe Lin, Haoxiang Li, and Bedrich Benes. It is currently being prepared for submission to ACM Transactions on Graphics (TOG). The first authors contribution includes implementation, testing, and manuscript preparation. Coauthors provided ideas, resources, and guidance in the research and helped in proofreading and modification in the manuscript preparation. Specifically, Yichen Sheng and Suren Deepak Rajasekaran helped with the creation of virtual training environments.

3.1 Abstract

Despite the rule-breaking masterpieces, general good photos follow empirical composition principles with the consideration of light, color, etc. The process of framing good photos can be highly automated. Previous work usually relies on heuristic rules or pre-knowledge of scenes, which make the applications challenging to scale onto real uncharted environments. Inspired by the human framing process, we set up a novel reinforced observation-action-reward solution for photography automation. We propose an intelligent mobile camera agent that autonomously seeks and composes aesthetically pleasing views for auto-detected regions of interest. We deploy the agent onto a high DoF drone after training it under photorealistic 3D virtual environments. The framework is able to scale to the real application, and the experiments show that Dr^{3}Cam outperforms heuristic baseline and acquires human-level framing on drone photography tasks in three testing scenes.

3.2 Introduction

"A good photograph is knowing where to stand." - Ansel Adams.

Photography is a visual art form. As with every visual art form, it is based on the elements and principles of art. The elements of art are the basic forms that an artist may use to construct an artwork, and it consists of *color, form, line, shape, space, texture*, and *value*. The principles of art are *balance, emphasis, movement, proportion, rhythm, unity*, and *variety*. The principals define ways of how an artist can organize the elements inside artwork that she/he intends to create (Gude, 2004).

In photography, the subject matter that we are trying to photograph are forms of these elements of art, such as the shape and silhouettes of the human, animal, or environmental subjects. Moreover, the composition of these elements and principles are based on compositional rules based on symmetry, golden ratio, phi grid, symmetry such as *Rule of Thirds, Rule of Odds, Rule of Space* and *Subframing* (Barnbaum, 2017).

Such principals can be computed (Liu et al., 2010) by algorithms, and further automate the photography (Zabarauskas & Cameron, 2014) with a mobile camera, that allows the camera orientation and position adjustments. Albeit researchers have explored several robot photographers (Byers et al., 2003; Myung-Jin Kim et al., 2010; Zabarauskas & Cameron, 2014), previous approaches are limited to simple camera Degree of Freedom (DoF) and monotonous composition rules. The previous rule-based methods are not flexible enough to take photos with arbitrary targets in a complex scene. Powered by recent Deep Learning (DL) research, Artificial Intelligence (AI) field has several significant breakthroughs. AIs approach or even surpass humans in several complex tasks, such as image recognition (He et al., 2016), video games (Mnih et al., 2015), and Go (Silver et al., 2016). General AI becomes a popular topic. David Silver asserts that: $Artificial\ Intelligence = Reinforcement\ Learning + Deep\ Learning$.

Think about how a human photographer takes a photo. A human photographer looks around the scene to find the Region of Interest (RoI) that draws attention. Then the photographer moves to a shooting spot to create proper distance and angle to the RoI. Considering the environment elements together with the RoI, the photographer fine-tunes the composition and eventually takes the photo.

Our key observation is that the human photography procedure can be fully automated with RL support that enables intelligent interactions with the environment. Therefore, we design an attention-based photography agent with the RL framework. Our 5-DoF camera agent is able to actively seek for surrounding RoI, autonomously adjust where it stands (or flies) by changing its position and orientation, and aesthetically composes visually appealing photos for the RoIs.

Moreover, we explore how well the agent can act as a real photographer in uncharted scenes by deploying it to a drone and conducting a user study. End-To-End drone training can be very time-consuming and even cause damages. Therefore, we utilize a photorealistic virtual training environment. However, a virtual environment can be problematic due to the gap between synthetic rendering and real-world scenarios (Zhang, Leitner, Milford, Upcroft, & Corke, 2015). We overstep the problem by novelly parsing the input view pixels into intermediate elements, such as convolutional features and salient information, as the observation. Our implementation of a drone manages to handle arbitrary scenes with multiple RoIs. The experiments show that our method outperforms heuristic method, and can achieve human-level photo framing with a drone.

In summary, our main contributions are:

- we propose a scalable RL solution for aesthetically pleasing view seeking and composing,
- we provide a novel framework for drone photography automation.

3.3 Related Work

We relate our work to the mobile camera control and reinforcement learning.

3.3.1 Mobile Camera Control

The virtual camera control has been widely studied in Computer Graphics, relating to the control of camera position, orientation, motion, and the more broader concepts such as viewpoint computation, motion planning, and editing (Christie & Olivier, 2009). Several early studies were conducted on object-based camera control assistance with single (Burtnyk, Khan, Fitzmaurice, Balakrishnan, & Kurtenbach, 2002; Khan, Komalo, Stam, Fitzmaurice, & Kurtenbach, 2005), and multiple targets (Christie, Normand, & Olivier, 2012). Lino and Christie (2012, 2015) suggested an algebraic approach to simplify the 6D camera searching space to a 2D manifold torus surface. The technique allows the virtual camera positioning for two or three target objects to be more efficient with no assumption on exact on-screen positioning. The work (Lino & Christie, 2012) was extended to guide the camera motion planning in crowd simulations (Galvane, Christie, Ronfard, Lim, & Cani, 2013), narrative-driven game (Galvane, Ronfard, Christie, & Szilas, 2014), and rail generation (Galvane, Christie, Lino, & Ronfard, 2015). These methods give flexible virtual camera control; however, they mostly rely on the pre-knowledge of the scene, which is not applicable in our situation.

Regarding real mobile camera control, researchers (Ahn et al., 2006; Byers et al., 2003; Campbell & Pillai, 2005; Myung-Jin Kim et al., 2010; Zabarauskas & Cameron, 2014) developed robot photographers, attaching camera(s) to a ground mobile robot, that can automatically navigate with collision avoidance and frame indoor photographs. Similar to our goal, the robot photographers actively seek for visually appealing views. However, the algorithms behind the robot photographers mostly depend on heuristic composition rules or similar techniques. The pure rule-based methods simply consider putting the salient features (e.g., faces) to

proper image-coordinate positions. This does not adequately account for complex scenes with arbitrary targets.

With the rapid development of drones, fly cameras are more accessible. Researchers dedicate to improve the drone manual control for photography (Huang, Yang, et al., 2018; Kang et al., 2018; Lan et al., 2017), design camera path with charted scenes (Gebhardt et al., 2016; Gebhardt, Stevšić, & Hilliges, 2018; Joubert et al., 2015; Roberts & Hanrahan, 2016; Xie et al., 2018), and apply autonomous cinematography on motion target (Bonatti, Yanfu, Choudhury, Wang, & Scherer, 2018; Fleureau et al., 2016; Galvane, Fleureau, Tariolle, & Guillotel, 2016; Galvane et al., 2018; Huang, Gao, et al., 2018; Huang, Lin, et al., 2019; Huang, Yang, et al., 2019; Nägeli et al., 2017; Ngeli et al., 2017). Previous work made in-depth contributions mainly to camera path optimization and target tracking. However, we do not see similar work to ours that enables the drone as an intelligent agent that actively explores the uncharted scene for acquiring impressive views.

3.3.2 Reinforcement Learning

Reinforcement learning (RL) framework allows agents to interact with the environment by taking actions basing on sensory observations (Arulkumaran, Deisenroth, Brundage, & Bharath, 2017; Sutton & Barto, 2018). In our case, the drone takes fly actions to adjust its pose to find an appealing composition for the region of interest based on the input camera view. Recent Deep Learning (DL) technique enables RL to perform human-level in decision-making tasks, such as video games (Mnih et al., 2015), Go (Silver et al., 2016) and driving (Kendall et al., 2018). RL implementations aim to improve the long-term outcomes and training efficiency with optimizations on value (Hasselt, Guez, & Silver, 2016; Mnih et al., 2015), policy (Schulman, Levine, Abbeel, Jordan, & Moritz, 2015; Schulman, Wolski, Dhariwal, Radford, & Klimov, 2017), or both (Mnih et al., 2016).

RL has been successfully applied to agent control (Gu et al., 2017; Lu et al., 2018; Won, Park, & Lee, 2018), and visual navigation (Gandhi, Pinto, & Gupta, 2017; Mo, Li, Lin, & Lee, 2018; Sadeghi & Levine, 2016; Tai, Paolo, & Liu, 2017; Zhu et al., 2017) problems which are relating to our work. Though there are end-to-end (Kahn, Villaflor, Pong, Abbeel, & Levine, 2017) training in the real environment for simple tasks, the majority of the training taken place in a virtual environment or with pre-collected offline data, this is because the trial-and-error training process of RL in the real world can be time-consuming or even cause damage.

To solve these problems, researchers begin to leverage synthetic methods to generate diverse training data. One mainstream hacks into commercial video games (Johnson-Roberson et al., 2017; Krahenbuhl, 2018; Richter, Hayder, & Koltun, 2017) to exploit art resources and high-quality rendering created by professional artists. However, hacking methods do not have adequate freedom in game controlling. Thus, another approach (Mueller, Casser, Lahoud, Smith, & Ghanem, 2017; Qiu et al., 2017) utilizes game engines for setting up a training environment, which enhances flexibility. Recent work by Shah, Dey, Lovett, and Kapoor (2018) introduces a physical-based photo-realistic simulator for training autonomous vehicles, that is beneficial to our study.

We embrace the virtual training environment due to data diversity and training efficiency. The primary concern of the virtual training environment is the over-fitting issue. The domain of synthetic visual features can be different from the real world scenario, which causes the trained model unable to scale to real world (Zhang et al., 2015). A natural solution is to improve photo-realistic rendering details (Shah et al., 2018; Zhu et al., 2017). Moreover, we also consider applying intermediate visual features (Section 3.6.1) as the observations instead of using raw pixels.

3.4 Overview

Our drone agent aims to take good photos autonomously. In this section, we first discuss the term of a "good photo" that our system considers in Section 3.4.1, and then give an introduction of the system blocks in Section 3.4.2.

3.4.1 A good photo

The empirical photography rules can definitely help in guiding photographers and algorithms for taking better pictures. However, based on human psychology and visual understanding of what makes an image aesthetically good, there are further complex levels in photography that makes a picture communicate on different levels with people. There are several photographs which broke these aforementioned compositional rules in producing better photos that are critically acclaimed.

These photographs with more profound meaning that is dependent on communicating with the audience in terms of lighting, composition, colors and subject matter are highly qualitative in nature, is currently not possible with automated methods as it requires more scene understanding, control, and intended audience. The current state of the art methods (Liu et al., 2010; Wei et al., 2018), and also ours, only allow us to focus **compositional correctness** at best, while considering color, light, and content as weak control factors that are hidden neurons in the networks.

3.4.2 System pipeline

The system is consisting of four major components, (1) the drone, (2) the mobile application, (3) the server, and (4) the virtual training environment. The blocks of the system can be seen in Figure 3.1. The drone streams its camera view in real-time to the mobile application (App) and sequentially executes moving commands (referring to actions in Section 3.6.2) sent back from the App. The App

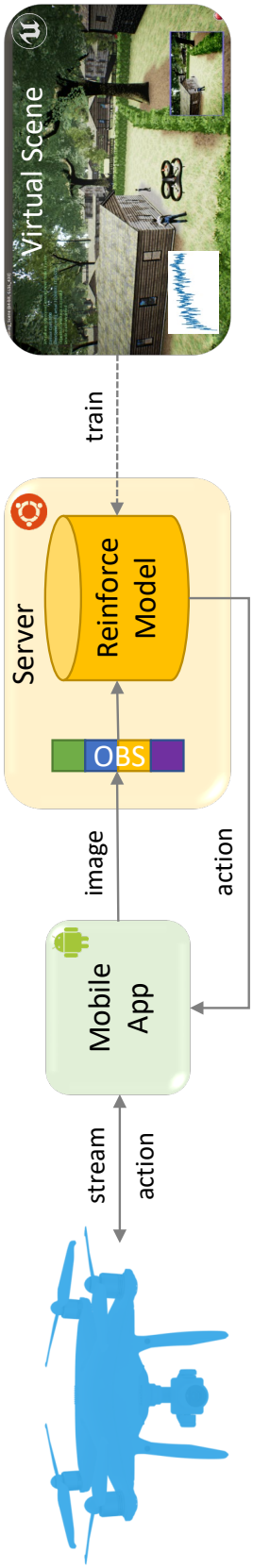

Figure 3.1.: The pipeline of the system.

synchronously communicates with the server by sending frames from the camera view stream and waits for the returning action responses from the server. Once each action response is retrieved from the server, the command is sent to the drone by the App. The server parses each frame into observations described in Section 3.6.1. The observations consist of several segments of intermediate visual features of the input frame, such as the convolutional features, salient information, view scores, etc. Each observation is passed into a pre-trained RL network that decides the agent next action. As a core segment of observation as well as the basis of rewards (referring to Section 3.6.3), the view scores are essential in predicting the camera agent actions for acquiring a good shooting position. We use a hybrid approach that combines a DL method with a computational method to score views discussed in Section 3.5. The neural network is trained in a photo-realistic virtual environment for minimizing the cost that is mentioned in Section 3.6.4. For more information about the RL implementation, please refer to Section 3.6.

3.5 View Scoring

Photos can be programmatic evaluated basing on composition principals, aesthetic characteristics, etc. Traditional computational methods emphasize on the compliance of the pre-defined rule conditions. The more recent data-driven approach, especially powered by Deep Learning (DL), train models to predict photo scores by utilizing image features. Our work aims to create a camera agent that can intelligently optimize its pose to hunt excellent views for photography. Therefore, scoring a view is one essential component of our study, which makes a part of the observation for the reinforced process (Section 3.6.1). Moreover, the optimization of RL aims at maximizing the rewards, which is formed by the view scores in our system (Section 3.6.3). We cover the pros and cons of DL, computational, and hybrid approaches in this section.

3.5.1 Deep Learning Approach

In our study, the views for each photography task come from the same shooting scene. The DL photo scoring models trained with non-relevant images (Kong, Shen, Lin, Mech, & Fowlkes, 2016; Murray, Marchesotti, & Perronnin, 2012) do not adequately consider the comparative nature of views sampled from the same scene. A ranking model trained with comparative same-scene-views is more suitable for our scenario. We adopt the siamese View Evaluation Net (VEN) from the state-of-the-art view composition work (Wei et al., 2018) to score the camera views.

The DL approach is scalable to general scenes. The VEN provides a satisfactory evaluation on complex views considering both foreground and background elements that empirical photography rule cannot merely apply to. However, the spatial score distribution is dispersed, and the VEN lacks having a preference for the salient objects. This makes the learning (Section 3.6) difficult the various dispersed good views cannot efficiently provide a pattern. The drone tends to make minor adjustments with few actions. Figure 3.2b offers a visualization of score distribution evaluated by VEN. Please also check the views and DL scores (the higher, the better) in Figure 3.3.

3.5.2 Computational Approach

Computational-based photo composition evaluation considers well-grounded composition guidelines, such as rule-of-thirds, diagonal dominance, triangular composition, visual balance, etc. Different from the DL (Section 3.5.1), the judging criteria are explicitly defined in the computational approach. The criteria work well with a specified RoI, that can help to compensate the insufficiency of the dispersed spatial score distribution with the DL approach. Our computational approach for the camera view scoring is inspired by (Liu et al., 2010). We focus on phi grid, visual balance, diagonal dominance, salient region size, and longest line placement.

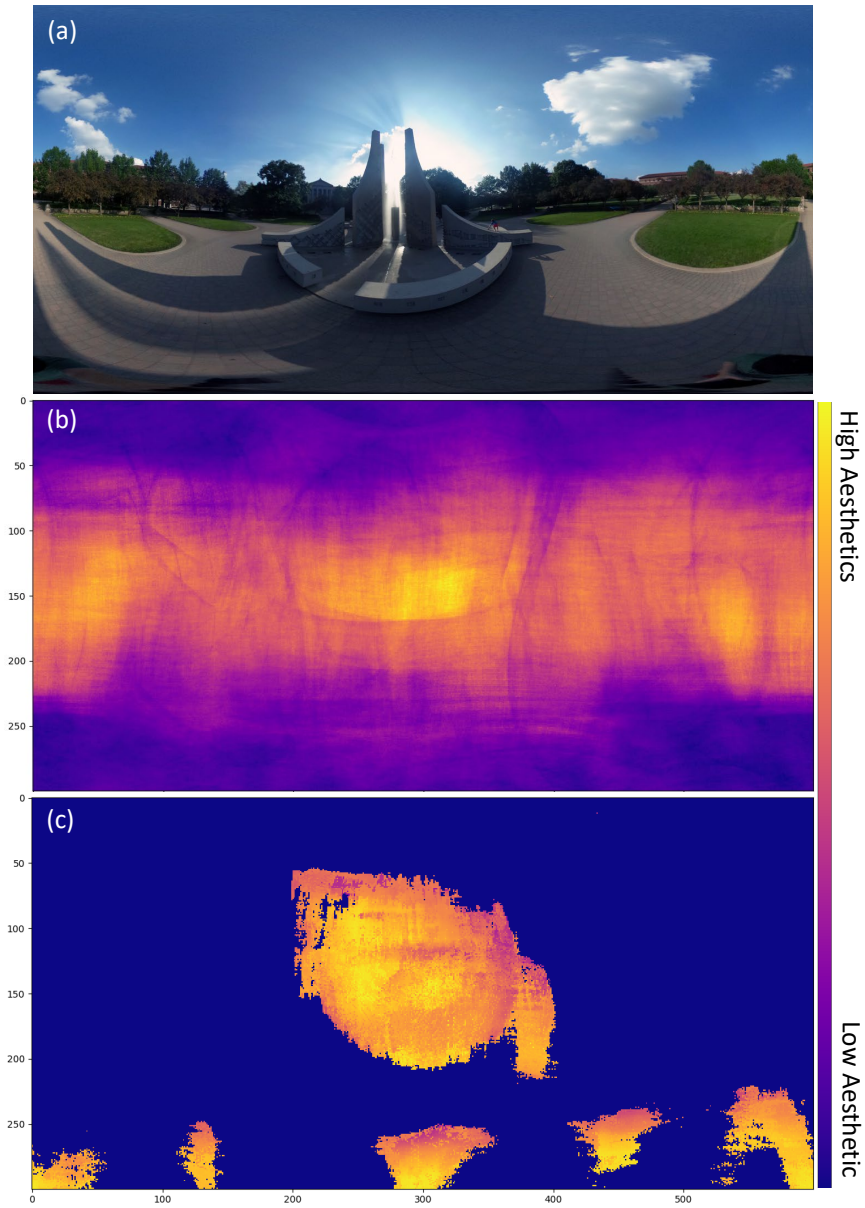

Figure 3.2.: The top row (a) is a flattened 360 spherical photo taken at one point of a scene. The rows (b) and (c) are the visualizations of score distribution with the DL approach (Section 3.5.1) and the computational approach (Section 3.5.2) . Each pixel in (b) and (c) represents the view score of a cropped and de-warpped sub-image centered by the same pixel coordinates in (a). The cropping window is with an 89° field-of-view and 512 × 384 resolution. The values of the score are mapped to colors - with yellow as the highest aesthetic and blue as the lowest. Please also refer to Figure 3.3, which shows the visualization of score distribution with the hybrid approach (Section 3.5.3) together with a few cropped view samples and scores predicted with the three approaches.

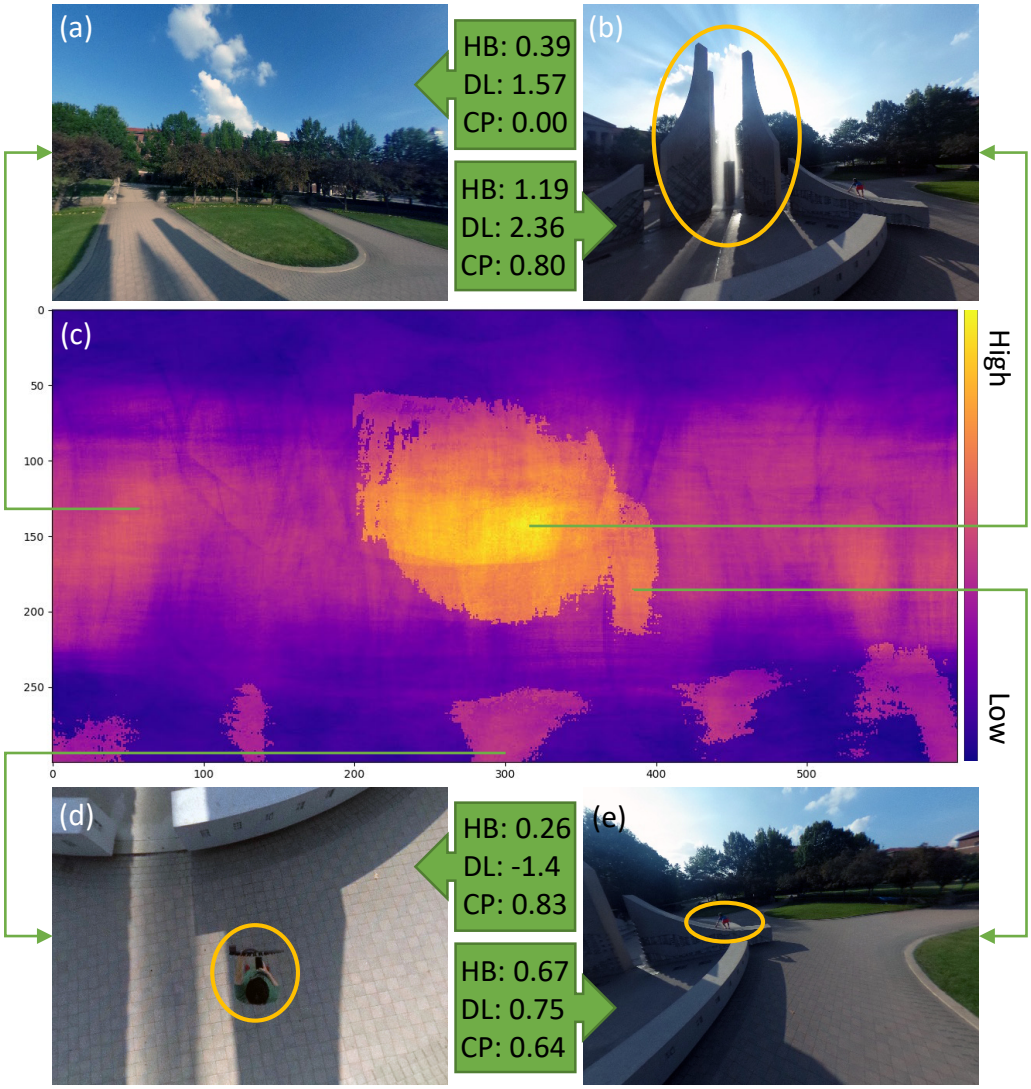

Figure 3.3.: The figure shows the visualization of score distribution (c) with the hybrid approach (referring to Section 3.5.3 and Figure 3.2), together with a few cropped and de-warpped view samples (a, b, d, and e) and their scores (in green arrow boxes) predicted with the three approaches. For the scores, HB stands for the hybrid approach (Section 3.5.3). DL stands for the deep learning approach (Section 3.5.1), and CP stands for the computational approach (Section 3.5.2). The RoIs (Hou et al., 2019) are circled out in the images.

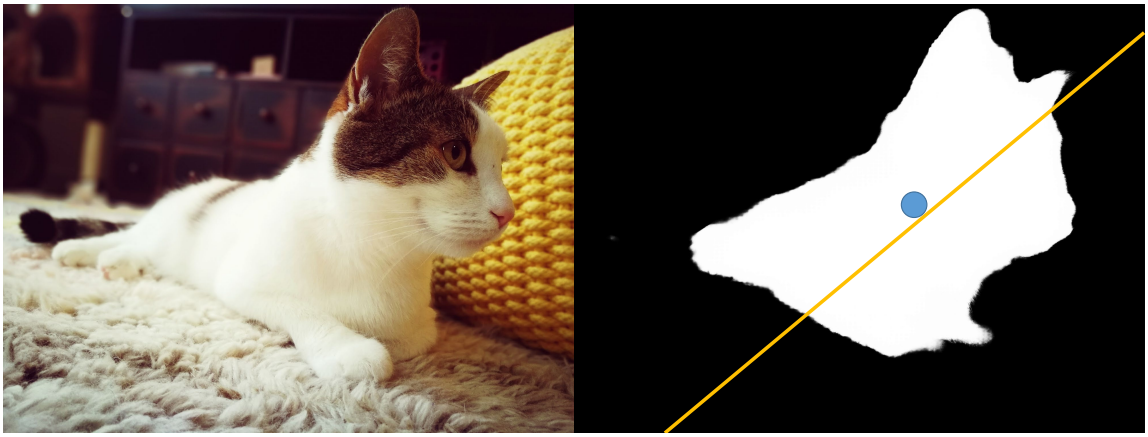

Figure 3.4.: The original photo (left) and the salieny map (right) with salient region detected with (Hou et al., 2019), centroid (blue), and principal axis (yellow).

In this section, *distance* refers to the *normalized Manhattan distance (NMD)*, which measures the distance between pair-wised pixel points or line segments in image coordinate suggested by Liu et al. (2010). The computational approach is visual attention based, which relies on salient-region information, such as salient centroids, principal axis, and region sizes, detected with the method proposed by Hou et al. (2019). A photo with its saliency map and corresponding information is demonstrated in Figure 3.4.

Phi grid

We use phi grid (Figure 3.5 red grid) to measure the position score of salient-regions of an image. Different from the similar rule-of-thirds grid (Figure 3.5 yellow grid), phi grid uses the golden ratio, which has a more precise mathematical definition. Moreover, phi grid has better scalability to wide photos. The equations to compute the phi grid score are shown in equations 3.1−3.5.

$$S_{phi} = \omega_p S_p + \omega_l S_l \tag{3.1}$$

We consider the phi grid score S_{phi} as a combination of a point score (Equation 3.2 and 3.3) and a line score (Equation 3.4 and 3.5). The point score S_p describes how close is each centroid of the salient region to the nearest grid line intersection. Similarly, the line score S_l describes how close is each principal axis of the salient region to the nearest grid line. The weight ω_p of point score S_p and the weight ω_l of line score S_l are set to 0.33 and 0.67.

$$S_p = \frac{\sum_{i=0}^{n} A_i \exp\left(-\frac{D_i^2}{\sigma_p}\right)}{\sum_{i=0}^{n} A_i} \tag{3.2}$$

$$D_i = \min_{j=1,2,3,4} d\left(C_i, I_j\right) \tag{3.3}$$

In the point score calculation above, A_i is the area size of each salient region. C_i and I_j are the centroids of salient regions and the four intersection points of the grid lines respectively. D_i in equation 3.3 represents the point NMD between each centroid and its closest intersection point.

$$S_l = \frac{\sum_{i=0}^{n} \exp\left(-\frac{D_i^2}{\sigma_l}\right)}{n} \tag{3.4}$$

$$D_i = \min_{j=1,2,3,4} d\left(L_i, L'_j\right) \tag{3.5}$$

In the line score calculation above, D_i in equation 3.5 represents the line-segment NMD between the principal axis and its nearest grid line for each salient region. L_i and L'_j respectively present the principal axis of each salient region and the four grid line segments. The line-segment NMD $d(L_i, L'_j)$ is defined as the average point NMD between all points of L_i and the closest points on L'_j. We experimentally set σ_p and σ_l to 0.04.

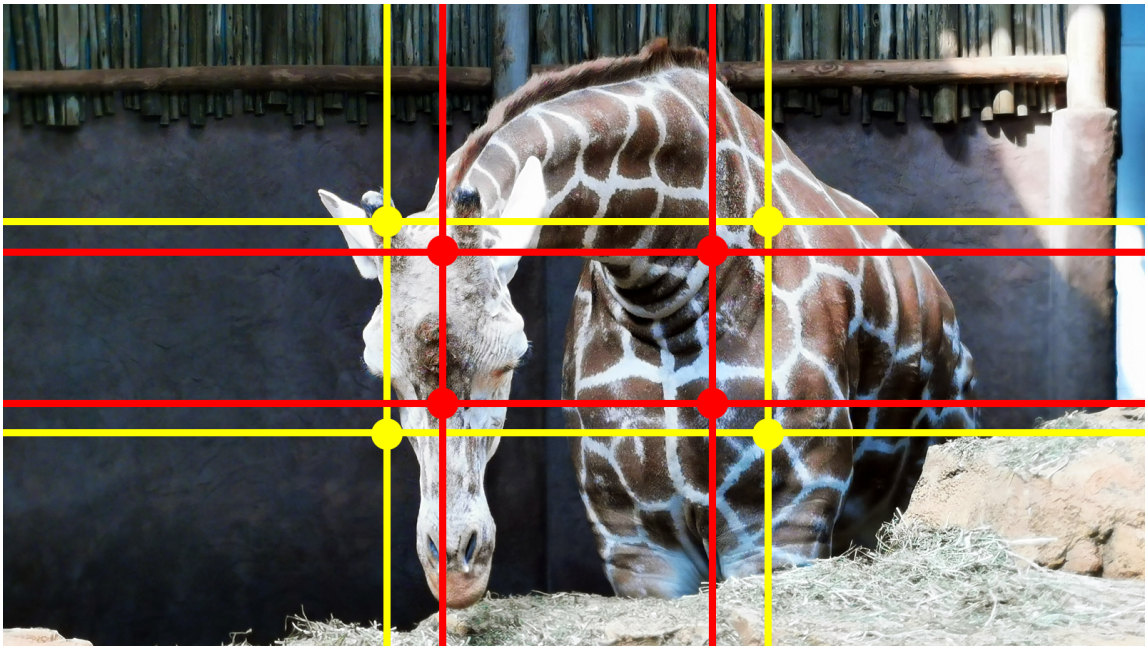

Figure 3.5.: Phi grid (red) and rule-of-thirds grid (yellow). The guidelines suggest to put salient region onto the grid line or intersection points.

Visual balance

Visual balance is another significant factor that determines the aesthetic of a photo. We use equation 3.6 and 3.7 to compute the score of visual balance.

$$S_{vb} = \exp\left(-\frac{D_i^2}{\sigma_{vb}}\right) \tag{3.6}$$

$$D_i = d\left(\frac{\sum_{i=0}^{n} C_i A_i}{\sum_{i=0}^{n} A_i}, C'\right) \tag{3.7}$$

In the visual balance score calculation above, C_i and C' respectively represent the centroid of each salient region and the image center. A_i is the area size of each salient region. D_i in equation 3.7 represents the point NMD between the area-weighted mean centroid of salient regions and the image center. We experimentally set σ_{vb} to 0.01.

Diagonal dominance

Diagonal dominance suggests the principal axis of a salient regions to be accord to either diagonal lines in the photo. The score can be computed similarly to the phi grid line score. The equations are shown below in equation 3.8 and 3.9.

$$S_{dd} = \frac{\sum_{i=0}^{n} \exp\left(-\frac{D_i^2}{\sigma_{dd}}\right)}{n} \tag{3.8}$$

$$D_i = \min_{j=1,2} d\left(L_i, L_j'\right) \tag{3.9}$$

In the diagonal dominance score calculation above, D_i in equation 3.9 represents the line-segment NMD between the principal axis and its nearest diagonal for each salient region. L_i and L_j' respectively present the principal axis of each salient region and the two diagonals. We experimentally set σ_{dd} to 0.1.

Salient region size

The previous study (Liu et al., 2010) pointed out that the ratio of salient region area size affects the aesthetics of a photo significantly. Especially in the case when we take photos of one salient object in different ranges with the same phi grid guidance. There are three significant high aesthetic peaks regarding the salient region area size ratios, which are around $r_1 = 0.1, r_2 = 0.56, r_3 = 0.82$. However, with our application on drones, it is uncommon to take a close shot that makes the salient region to occupy more than 80% of the photo. Therefore, we only consider the ratios $r_1 = 0.1$ and $r_2 = 0.56$. The equations for computing the salient-region sizes are shown below in equation 3.10.

$$S_{rs} = \max_{j=1,2} exp\left(-\frac{\left(\frac{\sum_{i=0}^{n} A_i}{A'} - r_j\right)^2}{\sigma_j}\right) \tag{3.10}$$

In the salient-region size score calculation above, A_i and A' respectively present the area size of each salient region and the image size in pixel. We experimentally set $\sigma_1 = 0.16$ and $\sigma_2 = 0.2$.

Longest line placement

The long lines such as a horizon line and the sea level line play an important role in photos, especially in aerial photos. Placing the long line horizontally, vertically to overlap with grid line such as rule-of-thirds lines makes the photo aesthetically pleasing. We detect the longest line (above 0.6 of image diagonal length if any) in the photo, and compute the longest line placement score with the equations 3.4 and 3.5 in Section 3.5.2. However, we replace the phi grid with the rule-of-thirds grid in the longest line placement score calculation to avoid the collision between the salient region and the longest line.

Combined score

We combine the weighted phi grid (S_{phi}), visual balance (S_{vb}), diagonal dominance (S_{dd}), salient region size (S_{rs}), and longest line placement (S_{ll}) scores together to form a final composition score for the photo with equation 3.11. We set $\omega_{phi} = 0.5, \omega_{vb} = 0.25, \omega_{rs} = 0.25, \omega_{dd} = 0.25$,and $\omega_{ll} = 0.25$. However, diagonal dominance is a nice-to-have guideline that does not necessarily to be existing in every high-aesthetic photo composition. Therefore, we put a threshold less than 0.8 to switch off diagonal dominance by making $\omega_{dd} = 0$ in such case. Similarly, the longest line may not exist in a photo, therefore, we make $\omega_{ll} = 0$ when no longest line is detected.

$$S_{cb} = \frac{\omega_{phi} S_{phi} + \omega_{vb} S_{vb} + \omega_{dd} S_{dd} + \omega_{rs} S_{rs} + \omega_{ll} S_{ll}}{\omega_{phi} + \omega_{vb} + \omega_{dd} + \omega_{rs} + \omega_{ll}} \tag{3.11}$$

Summary

In the computational approach, we apply several empirical principals that make a photo compositional pleasing. We formulate algorithms to examine the extent of the compliance on the salient regions or lines. This works well with photography tasks that have shooting targets. Figure 3.2c provides a visualization of score distribution evaluated by a rule-based approach. The high-aesthetic views are concentrated near the salient regions, such as the fountain and the persons. However, the rest views without salient object get score 0 regardless of the aesthetic level. This creates problems that cause the drone difficult to learn rotational actions (aircraft yaw, camera pitch) in the reinforce training (Section 3.6). We think this is because the views of a salient object in different angles with similar shape and size do not contribute significant score changes with the computational methods. Four view examples with scores are demonstrated in Figure 3.3; and the combined scores are noted with CP.

3.5.3 Hybrid Approach

Both of the previous approaches have their shortcomings. We combine the two methods in Equation 3.12 to form a new solution.

$$S_{hb} = \omega_{dl} S_{dl} + \omega_{cb} S_{cb} \tag{3.12}$$

Where, S_{dl} represents the score given by VEN in Section 3.5.1, and S_{cb} represents the combined score given by Equation 3.11 in Section 3.5.2. ω_{dl} and ω_{cb} are set to 0.25 and 0.75.

The hybrid approach takes into account the overall aesthetic of the view, but also the compliance of composition rules on the salient region. Figure 3.3c provides a visualization of score distribution evaluated by the hybrid approach. The scores of a few examples are demonstrated with hybrid scores (HB) in Figure 3.3. The learning (Section 3.6) with the hybrid approach results in better long-term camera pose optimization to form a high-aesthetic final composition in our experiments. We compare the final reinforced model trained with the hybrid and the computational approaches on the same target in Figure 3.6. We embrace the hybrid approach in our study.

3.6 Reinforcement Learning

Under the RL framework, an agent observes the environment then decides an action that maximizes the overall rewards. In our research, the drone agent sends back to the server its camera view image that is parsed into intermediate-level features as the observation (Section 3.6.1). The RL model consumes the observation to make an action from the pool (Section 3.6.2) that with confidence leads to high rewards defined in Section 3.6.3. We train the RL policy with value network (Section 3.6.4) with a photo-realistic virtual environment (Shah et al., 2018)

Figure 3.6.: One example of comparison between computational and hybrid approaches on a target object. Both experiments started at the same initial location. The hybrid approach performs more rotational actions.

discussed in Section 3.6.4, and deploy the model to our testing drone for real-world application described in Section 3.6.5.

3.6.1 Observation

We initially tested the pure pixel-to-action approach that takes the camera view image as the observation. The pixels are directly put into convolutional-based RL networks. However, this leads to poor convergence. Also considering the potential over-fitting issue with the virtual training environment in Section 3.6.4, we eventually choose to parse the camera view images into intermediate features as the observation.

The observation consists of four segments that are shown in Figure 3.7 left. We firstly extract a flattened $1280D$ feature vector as one observation segment with Mobilenetv2 (Sandler, Howard, Zhu, Zhmoginov, & Chen, 2018) considering the performance. We use a pre-trained ImageNet (Deng et al., 2009) 1000 classes classification model to enable learning transfer. An $11D$ one-hot vector of the previous action is constructed to be the second observation segment. Following the action vector, the RoIs of the image are detected (Hou et al., 2019) and encoded as bounding boxes to the third observation segment. The top three largest RoIs form a $12D$ vector with four negative ones fill-in for each missing RoI. The last observation segment is a 8 vector constructed with view scores in Section 3.5.2 in the form of $[S_{dl}, S_{vb}, S_{rs}, S_p, S_l, S_{dd}, S_{ll}, S_{hb}]$. The four segments are concatenated as a whole observation.

3.6.2 Actions

Our drone model has 5 Degree of Freedom (DoF) with 11 discrete actions shown in Figure 3.7 right. We consider 6 linear actions including *forward*, *backward*, *up*, *down*, *left* and *right*, 4 angular actions including *turn left*, *turn right*, *camera pitch up*, and *camera pitch down*, and one *stop* action.

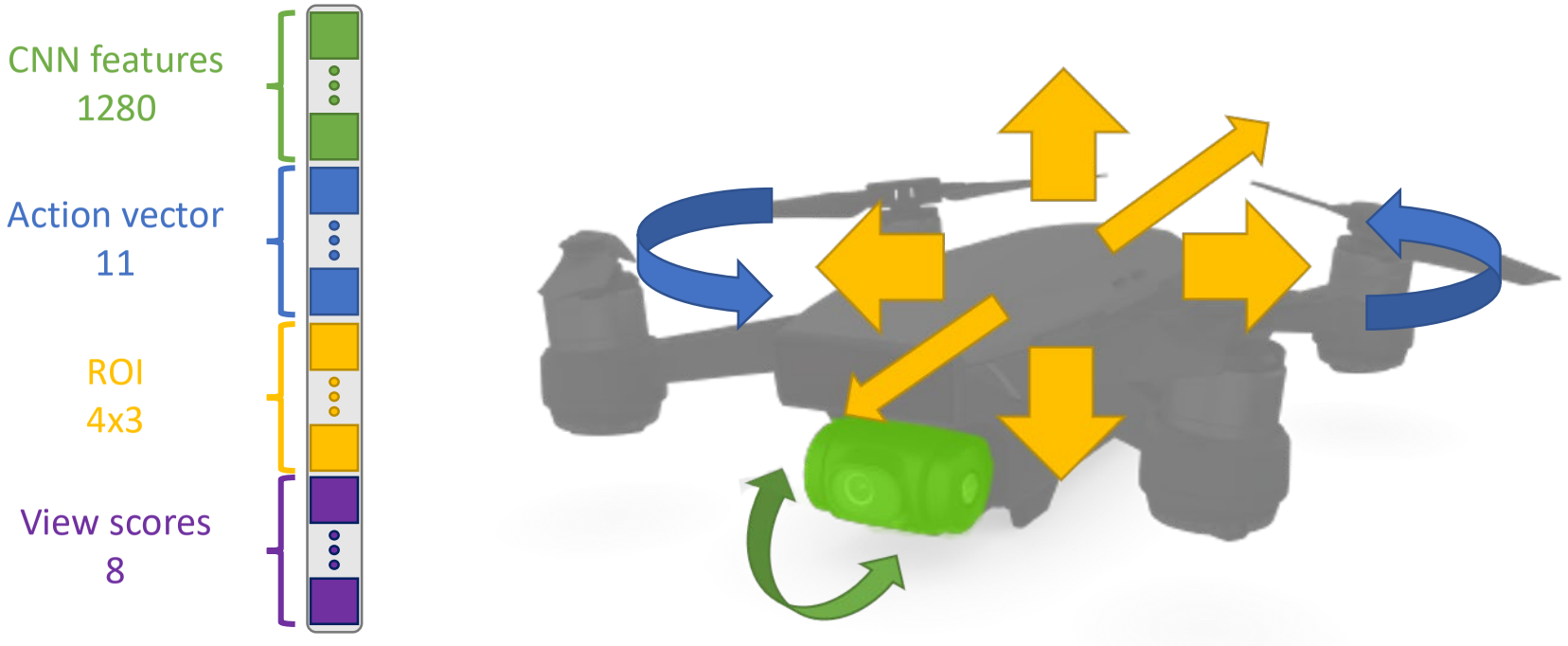

Figure 3.7.: The drone has 5 DoF with 11 discrete actions shown on the right beside a stop action. The drone camera view is parsed into four intermediate feature vectors concatenated as one observation shown on the left.

3.6.3 Reward Function

The reward is defined in Equation 3.13 and 3.14 below.

$$r_i = S_{hb}(i) - S_{hb}(i-1) - P \tag{3.13}$$

$$R = \sum_{i=0}^{n-1} r_i + cr_n \tag{3.14}$$

Where r_i indicates the reward for step i. $S_{hb}(i)$ represents the hybrid score (Section 3.5.3) of the camera view at step i. P is a constant that represents the penalty of an action. We set the action penalty to 0.005 for angular actions and 0.01 for the rest in our experiments. The accumulated reward R sums up the previous step rewards and finally adds the final step reward multiplied by a constant c that is larger than 1. We emphasis the value of the end step because the last camera view is the final outcome of our study that matters the most. c is set to 3 in our experiment.

3.6.4 Implementation and Training

We use the actor-critic implementation A2C [1], which is a synchronous deterministic variant of A3C (Mnih et al., 2016), for our task. Considering the time-series feature of the data, we apply Long-Short Term Memory (LSTM) (Hochreiter & Schmidhuber, 1997) architecture to our policy network.

We utilize a virtual training environment that is set up with Unreal Engine 4 [2] using AirSim (Shah et al., 2018) plugin. AirSim supports physics-based simulation with vehicles, including drones. We created two different virtual scenes for training. The first scene contains 10 car models. The cars are set to random orientations and positions for diversity during training. We trained the network for $1M$ action steps with the first scene, and then applied transfer learning to continue for another $1M$ steps on a larger neighborhood scene. The neighborhood scene

[1] https://openai.com/blog/baselines-acktr-a2c/
[2] https://www.unrealengine.com

Figure 3.8.: Virtual training environment.

contains more classes of objects, including persons, cars, houses, trees, traffic signs, etc. An overview is given in Figure 3.8.

3.6.5 Deployment

The system is consisting of three major components in our study: the drone, the Android application (app), and the server. A DJI Spark drone is used in our study. The drone streams its camera view in real-time to our app through DJI Mobile SDK [3], and executes *action* from the app. The app communicates with the server through ROS (Quigley et al., 2009) by sending a drone camera frame and receiving drone *action*. We deploy the RL model to a laptop as the server (CPU: Intel Core i7-7700HQ, GPU: NVIDIA GeForce GTX 1050 Ti, RAM: 16G). On the

[3]https://developer.dji.com/mobile-sdk/

server side, the node takes one frame each time and parses into an *observation*. The RL model on the server trades an *observation* with an *action* that is sent back to the app.

Our system supports two modes regarding RoI. The first mode (*Free Mode*) automatically detect salient region from drone camera view referring to Section 3.6.1 same as training. *Free Mode* fully enables the autonomous exploration of the scene. Though the final compositions are visually appealing (Section 3.7), the complete freedom makes the control of the shooting content difficult. However, it is useful for the situations that the users do not want to intervene in the searching process or do not have access to display the drone view. The second mode (*Target Mode*) is more restrict that takes user input rectangular selection from the app as RoI. This mode forces the drone to track a user-defined region and optimize its view around it, which is more suitable for photographic scenes with a clear target. In the *Target Mode*, our current system only takes one user-defined region and blocks the other salient detection. We put multiple users defined RoI *Target Mode* to the future work.

3.7 Evaluation

We evaluate the system on the drone by comparing aesthetic scores of photos taken by our method with the others made by several baseline methods, including three person experiments. The methods are listed below.

- *Randomness (RD)* The drone randomly samples actions (see Figure 3.7 left), and takes a photo when *Stop* is met.
- *Heuristic baseline (HR)* The drone detects the major salient target Hou et al. (2019), and applies the photograph composition heuristics used by Byers et al. (2003); Zabarauskas and Cameron (2014) including: *Rule of thirds, No middle rule, No edge rule*, and *Occupancy rule* on to the target.

- *Person* $1-3$ *(P*1-*P*3*)* Three persons manually operates the drone to take photos with DJI GO 4 app [4]. All three persons rate their photography skill to be above average.

- *Ours* The drone uses our intelligent camera agent to take photo fully autonomously with *Free Mode*.

We chose three different scenes (school, bell tower, and neighborhood). At each scene, we fixed five starting points and took four photos starting from the point with the six methods. In total, we got $4 \times 5 \times 3 \times 6 = 360$ photos. The photos were scored as Human Intelligence Tasks (HITs) with Mechanical Turk [5]. The instruction (Kong et al., 2016) is described as *Please rate the photo w.r.t its aesthetic. This is a 5-scale choice, from 1 through 5, meaning from low to high aesthetic.* The 5-scales and their corresponding scores are: *Very Bad (*1*), Bad (*2*), Neutral (*3*), Good (*4*)*, and *Very Good (*5*)*. 20 workers scored each photo. The proportions of photo ratings with different methods are shown in Figure 3.9. Our method is closest to and slightly better than human framing (P1, P2, and P3) in the experiments compared to the other two baselines. We also verify the hypothesis with statistic tests.

Because of the 5-scale ranking, the variables are ordinal. Non-parametric Mann-Whitney U tests were conducted on the Likert item data. The mode of the 20 rankings from HITs is used for each photo. We set null hypothesis H_0 being: *the mean ranks of the two groups are equal.* Table 3.1 shows that there are significant differences between the mean ranks of the rating of photos taken by RD and humans. The test statistics also imply that HR baseline performed similarly to P2 and P3, and our method does not result in significant mean rank differences with human operations in the experiments.

We also computed the average rating score basing on the $1-5$ ratings of the 20 HITs for each photo and further conducted parametric ANOVA tests on the

[4] https://www.dji.com/downloads/djiapp/dji-go-4
[5] https://www.mturk.com/

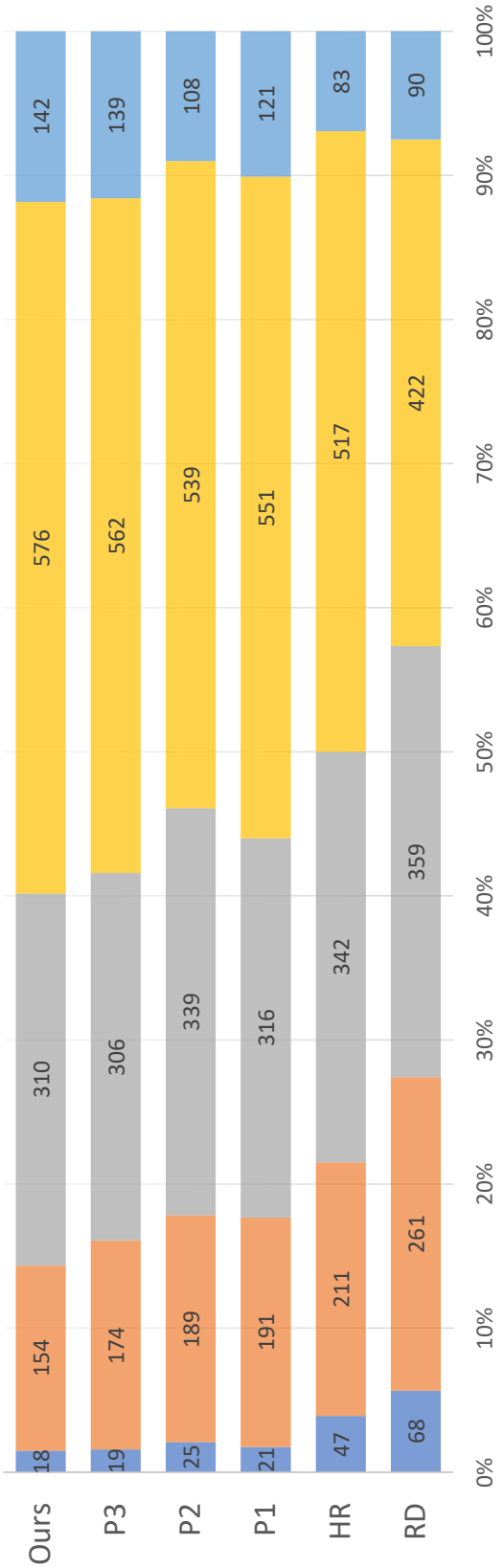

Figure 3.9.: The proportions of photo ratings with different approaches.

Table 3.1.: Mann-Whitney U Test statistics on autonomous method versus human operation pairs. Non-statistically significant pairs ($\alpha = 0.05$) are marked in orange, indicating the failing rejection of H_0, and meaning that the mean ranks of the two groups are statistically equal.

U Test	P1	P2	P3	P1-P3
RD	$W = 1266$ $n_1 = 60$ $n_2 = 60$ $P < 0.001$	$W = 1373$ $n_1 = 60$ $n_2 = 60$ $P = 0.010$	$W = 1295$ $n_1 = 60$ $n_2 = 60$ $P = 0.002$	$W = 3935$ $n_1 = 60$ $n_2 = 180$ $P < 0.001$
HR	$W = 1478$ $n_1 = 60$ $n_2 = 60$ $P = 0.037$	$W = 1592$ $n_1 = 60$ $n_2 = 60$ $P = 0.191$	$W = 1501$ $n_1 = 60$ $n_2 = 60$ $P = 0.054$	$W = 4572$ $n_1 = 60$ $n_2 = 180$ $P = 0.025$
Ours	$W = 1831$ $n_1 = 60$ $n_2 = 60$ $P = 0.821$	$W = 1947$ $n_1 = 60$ $n_2 = 60$ $P = 0.307$	$W = 1843$ $n_1 = 60$ $n_2 = 60$ $P = 0.753$	$W = 5621$ $n_1 = 60$ $n_2 = 180$ $P = 0.525$

processed data. The null hypothesis H_0 is set to *the mean score is the same for the groups*. Similar to the Mann-Whitney U tests, the AVONA statistics are shown in Table 3.2, and lead to similar conclusions to the non-parametric tests. The data show that the mean score of photos taken by our method does not significantly differ from human captures in our experiments.

Example photos of the experiments with our method are shown in Figure 3.10. The first column shows two example photos rated as *Good*, the second column for *Neutral*, and the last column for *Bad*.

3.8 Conclusion

We have introduced an intelligent mobile agent that can seek and compose good views automatically. The agent has been deployed to a drone for photography automation. Though the system can compose user-satisfied photos in the experiments, there are several potential limitations.

Figure 3.10.: Example photos from the experiments with our method. HITs indicate the mode of the 20 rankings for each photo from MTurk survey. HB scores show the score prediction with our hybrid approach described in Section 3.5.3.

Table 3.2.: ANOVA test statistics on autonomous method versus human operation pairs. Non-statistically significant pairs ($\alpha = 0.05$) are marked in orange, indicating the failing rejection of H_0, and meaning that the mean scores are statistically the same for the comparing groups.

ANOVA	P1	P2	P3	P1-P3
RD	$F(1, 118)$ $= 14.757$ $P < 0.001$	$F(1, 118)$ $= 12.347$ $P < 0.001$	$F(1, 118)$ $= 24.257$ $P < 0.001$	$F(3, 236)$ $= 10.018$ $P < 0.001$
HR	$F(1, 118)$ $= 4.011$ $P = 0.047$	$F(1, 118)$ $= 2.524$ $P = 0.115$	$F(1, 118)$ $= 8.811$ $P = 0.004$	$F(3, 236)$ $= 3.251$ $P = 0.0235$
Ours	$F(1, 118)$ $= 1.630$ $P = 0.204$	$F(1, 118)$ $= 3.533$ $P = 0.063$	$F(1, 118)$ $= 0.282$ $P = 0.597$	$F(3, 236)$ $= 1.466$ $P = 0.224$

The drone uses passive obstacle avoidance, that does not continue the task after the detection of obstacle within a close range. The autonomous camera agent decides its motions basing on the RoIs. However, if the user does not select the RoIs, the automatic saliency detection may lead to poor RoIs that do not contain human interested features. The improvement requires a further scene understanding mechanism. Also, the terms (e.g., phi-grid, visual balance) that are used in judging the photo composition and aesthetic levels are limited; therefore, we can explore more photographic terms and combinations that make a better view scoring standard in the future work. Moreover, the weights of the terms used for the reward function are set experimentally, which may not be the most optimized selection. Our solution on the good view seeking may lead to local maximum points; however, how to find the global best in an uncharted scene will be another exciting direction.

CHAPTER 4. CONCLUSIONS AND FUTURE WORK

Coming back to the original research question: *Can we improve the view composing process with a mobile camera?* We came up with a two-fold solution, basing on which we proposed two frameworks that have been successfully applied to drones for photography tasks. On the one hand, we introduced a novel drone HRI method that enables intuitive goal-oriented drone photography. On the other hand, we trained an intelligent drone agent that actively seeks and composes visually pleasing photos automatically. Both sides aim at the improvement of camera view composition process. In this chapter, we make conclusions about our work. We also discuss the limitations of our system and potential future work in this direction.

4.1 Multi-touch Gesture Controlled Drone Gimbal Photography

Controlling a high DoF mobile camera, including a drone, is complicated. Operators have to apply continuous eye-hand coordination on the camera for pose changes, which creates challenges in high-level view composing tasks while keeping low level camera movements in mind all the time. For example, the typical control of a drone is through a dual joystick RC. In order to take a desired photo, the user must steer the drone to reach an estimated location, then adjust camera orientation to see the resulting view. If the view is not as expected, the drone needs to be moved further, camera adjusted, etc. This process can happen several times before the view is reached. We asked two questions regarding the situation.

- *Is it possible to make the drone HRI more intuitive and natural?*
- *How to let the user focus on photography tasks without thinking about drone motor controls?*

For the first question, we decouple the flight from the camera operations. We provide a unified framework that encapsulates control of the movement of drones and camera control. In this way, the user does not need to think about the drone movement and camera control as two separate actions naturally, she/he can focus on the photography tasks. One subject left the feedback after the user study: *"the camera-pointing-forward function provided by the tablet only mode (FlyCam) is super useful. (It is) the must-have function!"* For the second question, we designed six simple touch gestures. The user can apply gestures onto a touch-screen mobile device that is the replacement of the RC. Another subject mentioned: *"the gesture method (FlyCam) is new to me, and I was less skilled at using it to control the drone, but it was relatively easier to learn."* Moreover, the gestures are conducted on top of the real-time drone camera streaming, which makes the user feel like manipulating the camera view directly. We speculated that the FlyCam HRI could lead to the easiness of navigation and efficiency in drone photography tasks. The 20 human subjects user study proved that FlyCam is more intuitive and with a lower workload than the traditional RC method.

4.2 Limitations of Multi-touch Gesture Controlled Drone Gimbal Photography

FlyCam has several *limitations*. First, the communication between the drone and the mobile device has a delay, which causes control less responsive. A user pointed out in the feedback: *"I preferred the controller more as well as how responsive the controls were."* However, the future hardware update will eventually minimize the delay. Second, the tap-hold gesture (moving forward) is not a natural gesture for photography tasks, potentially caused by the conventional mouse double-click behavior. Moreover, the delay threshold of the tap-hold and double tap hold gestures may lead to proximal drone adjustment not smooth. Zoom in and out pinch could be a better replacement to the tap-hold gesture, as it is widely used in camera/map applications.

4.3 Region-of-Interest Based Reinforced Drone Photography

Our second starting point was to simplify the mobile camera view composing process. Considering the drone as the carrier, we formed the research question:

Is it possible to simplify the drone photography process with aesthetic composition consideration?

Previous work (Xie et al., 2018) focuses more on trajectory optimizations in a charted environments. Some other approaches (Zabarauskas & Cameron, 2014) utilize rule-based composition guidance to make the low DoF photography automation, which does not apply to complex targets and scenes. Therefore, we further asked the question:

Is it possible to fully automate drone photography for uncharted scenes?

Inspired by the real photographer behaviors, we observed that the photography procedures could be encapsulated into RL settings, as an observation-action-reward process. We surmised that we could make a camera agent that aims at finding RoIs and compose pleasing views through the trial-and-error training. We could deploy such an agent onto a drone for photography automation. However, considering the end-to-end training cost and risk, we utilized a virtual training environment. The system novelly concatenates intermediate visual features as the observation to overcome the gap between the virtual environment and the real-world scene. The experiment results show that our drone photography automation method outperformed the heuristic baseline and achieved human-level composing standard.

4.4 Limitations of Region-of-Interest Based Reinforced Drone Photography

Dr^{3}Cam also has several *limitations*. First of all, we used a hybrid view scoring method that contains the rule-based consideration. We take into account of *phi grid, visual balance, diagonal dominance, region size*, and *longest line placement*; however, the limited term selection may cause the ignore for other aspects. Further,

we experimentally set the weights of rule-based terms in the reward calculations, which may not lead to the most optimized solution. Therefore, further investigations on more photographic rule terms with a different combination of weights is an interesting follow-up study. Second, we do not take into consideration of active OA, which is a good-to-have feature in the drone automation tasks. But we think it is more about the sensor enhancement of drone hardware industry. Third, though it has been proved that our method can lead to a visually appealing view composing, it does not guarantee that the final shot is the global maximum view in the scene. It is a promising topic to explore the most efficient way to approach to the best view.

4.5 Future Work

The current digital cameras usually support both the manual mode and the auto mode, and both modes cannot be removed. We believe the mobile camera view composing, including drone photography, is with the same situation as the two-mode digital cameras. Users need both manual control and automation method for different tasks. We also consider two directions for the future work.

For the manual controls, there are several potential directions. Starting with our FlyCam framework, we intentionally designed the minimal number of gestures to keep the system simple. However, it could be possible to extend the gesture pool for more complex photography commands, such as orbiting a target. The exploration of broad touch gestures for drone photography tasks is a promising topic. Moreover, body gesture control on simple tasks, such as triggering the shutter or tracking a subject, has already been employed to consumer drones. While the accurate interactions on RoI specification and drone pose fine adjustments are still not satisfactory on photography tasks, but very much required as a natural interaction method. A further investigation of the accurate HDI with body gestures needs to be conducted.

Since drones are also widely used to take video footage, while many existing works dedicate to the trajectory design and optimization (Gebhardt et al., 2018; Joubert et al., 2015; Roberts & Hanrahan, 2016) and neglect the real-time interactivity of the drone control. We can expand the multi-touch solution to drone cinematography, which changes the focus from photography to the videography tasks. It could be challenging as well as valuable to seek for a responsive solution that enables the users to intuitively change the drone path while complying with the C^4 continuity (Roberts & Hanrahan, 2016) of the trajectory.

Furthermore, AI algorithms could give exceptional support to our multi-touch gesture-controlled solution. For example, if the user points at a target and the target can be automatically selected and segmented out, the user could drag and drop the selection on the screen directly to where she/he wants to place the target in the photo. The selection could be more helpful if advanced 3D reconstruction is applied to the target, so that the user can manipulate the 3D model of the target on the screen with touch gestures, to decide from where the camera should look at the target. The assistance of view composing supported with neural networks could be an extension of the FlyCam.

Considering the HRI, we also believe coordination is an interesting problem. We divide the coordination into two cases: multiple users controlling the same drone and one user controlling multiple drones. Basing on the FlyCam framework, one user can control the drone entirely from a single mobile device. What if two users use two mobile devices to connect to the drone at the same time and want to take photos together? We need to study the user coordination behaviors further to give out a reasonable design. On the other hand, users may also want to take photos of the same subject from different angles/ranges, and it could be more efficient if there were an intuitive method to operate multiple drones at the same time on one task. Multiple drone control with a single mobile device using touch gesture could be another potential area of future work for our study.

One critical feature of our framework is the low workload and pressure. We introduced FlyCam for drone photography tasks; however, the touch gesture based goal-oriented design could have potential use in other high DoF remote controls such as robot arm manual manipulation. It could be excited to migrate the design onto different tasks other than photography.

For automation, there are more interesting follow-up topics. As mentioned in Section 4.3, local maximum reward that leads to the early termination of searching is a potential limitation of Dr^{3}Cam. Therefore, we develop a further research question: *Will the global maximum reward lead to a significantly different result?* We put this to future work because we believe that it is not possible to find the global best view within a reasonable time in an uncharted scene with current settings. We would like first to justify the research question. If we can confirm that the global maximum reward leads to significantly better compositions in photography tasks, we may continue the exploration in this area. With better scene understanding techniques, it could be a promising avenue to continue seeking global best view solutions.

For the photography automation in uncharted scenes, it could be helpful if we can have a sort of understanding of the context. Therefore, advanced ML algorithms such as object detection, semantic segmentation, and target tracking may contribute improvements to our study. Moreover, it is interesting to investigate the training process of RL with various implementations, which may give us different convergence efficiency and accumulated reward that lead to potential enhancement of the model. We may further investigate the utilization of additional sensors, such as depth and 360 cameras. For example, the depth camera can be used to retrieve additional 3D information of the salient object, which could be beneficial in long-term camera motion planning. The 360 can provide a panorama view of the scene, which could make the search of RoIs more flexible and accurate, and it is potential support to seek for the global best view.

Another exciting direction is to learn the user composition style and apply the view composing automation considering the style. With many pre-taken photos of one user, it is possible to determine the preference of the user. By redefining the reward functions according to the user style, it could be possible to develop a personalized drone photographer. Along this direction, it could also be valuable to study the styles of drone cinematography trajectories. Basing on the professional drone videos, it could be possible to study the drone actions inversely. The continuous actions make a styled trajectory. By successfully transferring a styled professional trajectory to an arbitrary trajectory, we may get excellent cinematography.

Multiple drones coordination is another potential future research area. Similar to the point discussed in HRI future work, it could lead to better efficiency if multiple drones could communicate with each other for acquiring a variety of excellent views inside a scene at the same time. Moreover, it could offer a potential high-performance solution to seek for the global best view of an uncharted scene. Further, the automation of multiple drone coordination could contribute to broader fields such as scene scanning and reconstruction, searching and rescue, etc.

Since end-to-end training is not suitable in many training tasks, virtual environment or synthetic data generation is a trending area that needs exploration. However, the well-known over-fitting issue caused by the difference between training and testing scenes is a primary concern. Therefore, it is excited to conduct an in-depth study of the artificial features that may cause the gap. It could be beneficial for many AI related field including autonomous driving and computer vision.

Last but not least, we set up an RL framework with virtual training environments, and automate the drone photography by setting a view-score-based reward function. However, the pipeline may be used for other similar tasks by replacing the reward function and the virtual training environment. For example, 3D scanning automation with drones could be a potential investigation direction.

4.6 Summary

In this dissertation, we introduced two drone photography frameworks that respectively account for natural HRI and full automation. As the carrier of the mobile camera view composing study, we expect our methods to bring new ideas to the robotics community as well as make initial attempts in general AI field. We hope to continue working in this direction and contributing to the brighter future of technology.

LIST OF REFERENCES

LIST OF REFERENCES

3D-Robotics. (2017, March). Solo user manual v9.02.25.16 [Computer software manual]. (https://3dr.com/solo-drone)

Abtahi, P., Zhao, D. Y., E., J. L., & Landay, J. A. (2017, September). Drone near me: Exploring touch-based human-drone interaction. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, *1*(3), 34:1–34:8. Retrieved from http://doi.acm.org/10.1145/3130899 doi: 10.1145/3130899

Ahn, H., Kim, D., Lee, J., Chi, S., Kim, K., Kim, J., ... Kim, H. (2006, Oct). A robot photographer with user interactivity. In *2006 ieee/rsj international conference on intelligent robots and systems* (p. 5637-5643). doi: 10.1109/IROS.2006.282286

Arulkumaran, K., Deisenroth, M. P., Brundage, M., & Bharath, A. A. (2017, Nov). Deep reinforcement learning: A brief survey. *IEEE Signal Processing Magazine*, *34*(6), 26-38. doi: 10.1109/MSP.2017.2743240

Bangor, A., Kortum, P., & Miller, J. (2009). Determining what individual sus scores mean: Adding an adjective rating scale. *Journal of usability studies*, *4*(3), 114–123.

Barnbaum, B. (2017). *The art of photography: A personal approach to artistic expression*. Rocky Nook, Inc.

Bonatti, R., Yanfu, Z., Choudhury, S., Wang, W., & Scherer, S. (2018, November). Autonomous drone cinematographer: Using artistic principles to create smooth, safe, occlusion-free trajectories for aerial filming. In *International symposium on experimental robotics*. Springer.

Brooke, J. (1996). Sus-a quick and dirty usability scale. *Usability evaluation in industry*, *189*(194), 4–7.

Buchanan, S., Floyd, B., Holderness, W., & LaViola, J. J. (2013). Towards user-defined multi-touch gestures for 3d objects. In *Proceedings of the 2013 acm international conference on interactive tabletops and surfaces* (pp. 231–240). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/2512349.2512825 doi: 10.1145/2512349.2512825

Burtnyk, N., Khan, A., Fitzmaurice, G., Balakrishnan, R., & Kurtenbach, G. (2002). Stylecam: Interactive stylized 3d navigation using integrated spatial & temporal controls. In *Proceedings of the 15th annual acm symposium on user interface software and technology* (pp. 101–110). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/571985.572000 doi: 10.1145/571985.572000

Byers, Z., Dixon, M., Goodier, K., Grimm, C. M., & Smart, W. D. (2003, Oct). An autonomous robot photographer. In *Proceedings 2003 ieee/rsj international conference on intelligent robots and systems (iros 2003) (cat. no.03ch37453)* (Vol. 3, p. 2636-2641 vol.3). doi: 10.1109/IROS.2003.1249268

Campbell, J., & Pillai, P. (2005, April). Leveraging limited autonomous mobility to frame attractive group photos. In *Proceedings of the 2005 ieee international conference on robotics and automation* (p. 3396-3401). doi: 10.1109/ROBOT.2005.1570635

Cauchard, J. R., E, J. L., Zhai, K. Y., & Landay, J. A. (2015). Drone & me: An exploration into natural human-drone interaction. In *Proceedings of the 2015 acm international joint conference on pervasive and ubiquitous computing* (pp. 361–365). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/2750858.2805823 doi: 10.1145/2750858.2805823

Cavalcanti, C., Gomes, H., Meireles, R., & Guerra, W. (2006). Towards automating photographic composition of people. In *Proceedings of the iasted international conference on visualization, imaging, and image processing* (pp. 25–30).

Chandarana, M., Trujillo, A., Shimada, K., & Danette Allen, B. (2017). A natural interaction interface for uavs using intuitive gesture recognition. In P. Savage-Knepshield & J. Chen (Eds.), *Advances in human factors in robots and unmanned systems* (pp. 387–398). Cham: Springer International Publishing.

Chen, Y.-L., Lee, W.-T., Chan, L., Liang, R.-H., & Chen, B.-Y. (2015). Direct view manipulation for drone photography. In *Siggraph asia 2015 posters* (pp. 23:1–23:1). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/2820926.2820945 doi: 10.1145/2820926.2820945

Christie, M., Normand, J.-M., & Olivier, P. (2012). Occlusion-free camera control for multiple targets. In *Proceedings of the acm siggraph/eurographics symposium on computer animation* (pp. 59–64). Goslar Germany, Germany: Eurographics Association. Retrieved from http://dl.acm.org/citation.cfm?id=2422356.2422366

Christie, M., & Olivier, P. (2009). Camera control in computer graphics: Models, techniques and applications. In *Acm siggraph asia 2009 courses* (pp. 3:1–3:197). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/1665817.1665820 doi: 10.1145/1665817.1665820

Christie, M., Olivier, P., & Normand, J.-M. (2008). Camera control in computer graphics. *Computer Graphics Forum*, *27*(8), 2197-2218. Retrieved from https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-8659.2008.01181.x doi: 10.1111/j.1467-8659.2008.01181.x

Crescenzio, F. D., Miranda, G., Persiani, F., & Bombardi, T. (2009, June). A first implementation of an advanced 3d interface to control and supervise uav (uninhabited aerial vehicles) missions. *Presence*, *18*(3), 171-184. doi: 10.1162/pres.18.3.171

Deng, J., Dong, W., Socher, R., Li, L., Kai Li, & Li Fei-Fei. (2009, June). Imagenet: A large-scale hierarchical image database. In *2009 ieee conference on computer vision and pattern recognition* (p. 248-255). doi: 10.1109/CVPR.2009.5206848

DJI. (2017a, August). Dji goggles user guid v1.2 [Computer software manual]. (`https://www.dji.com/dji-goggles`)

DJI. (2017b). *Dji mobile sdk for android.* Retrieved from `https://github.com/dji-sdk/Mobile-SDK-Android`

DJI. (2017c). *Dji uilibrary for android.* Retrieved from `https://github.com/dji-sdk/Mobile-UILibrary-Android`

DJI. (2017d, July). Phantom 4 pro/pro+ user manual v1.6 [Computer software manual]. (`https://www.dji.com/phantom-4-pro`)

DongBin Lee, Vilas Chitrakaran, Burg, T., Dawson, D., & Bin Xian. (2007, Dec). Control of a remotely operated quadrotor aerial vehicle and camera unit using a fly-the-camera perspective. In *2007 46th ieee conference on decision and control* (p. 6412-6417). doi: 10.1109/CDC.2007.4434940

E, J. L., E, I. L., Landay, J. A., & Cauchard, J. R. (2017). Drone & wo: Cultural influences on human-drone interaction techniques. In *Proceedings of the 2017 chi conference on human factors in computing systems* (pp. 6794–6799). New York, NY, USA: ACM. Retrieved from `http://doi.acm.org/10.1145/3025453.3025755` doi: 10.1145/3025453.3025755

Ehang. (2016, December). Ghostdrone 2.0 ehang play app manual [Computer software manual]. (`http://www.ehang.com/ghost2.0.html`)

Ettikkalayil, J. M. (2013). *Design, implementation, and performance study of an open source eye-control system to pilot a parrot ar. drone quadrocopter.* Unpublished master's thesis, City University of New York.

Fernndez, R. A. S., Sanchez-Lopez, J. L., Sampedro, C., Bavle, H., Molina, M., & Campoy, P. (2016, June). Natural user interfaces for human-drone multi-modal interaction. In *2016 international conference on unmanned aircraft systems (icuas)* (p. 1013-1022). doi: 10.1109/ICUAS.2016.7502665

Fiorella, D., Sanna, A., & Lamberti, F. (2010, Mar 01). Multi-touch user interface evaluation for 3d object manipulation on mobile devices. *Journal on Multimodal User Interfaces*, *4*(1), 3–10. Retrieved from `https://doi.org/10.1007/s12193-009-0034-4` doi: 10.1007/s12193-009-0034-4

Fleureau, J., Galvane, Q., Tariolle, F.-L., & Guillotel, P. (2016). Generic drone control platform for autonomous capture of cinema scenes. In *Proceedings of the 2nd workshop on micro aerial vehicle networks, systems, and applications for civilian use* (pp. 35–40). New York, NY, USA: ACM. Retrieved from `http://doi.acm.org/10.1145/2935620.2935622` doi: 10.1145/2935620.2935622

Forbes, T. (2019, July). *The art of photography.* Retrieved from http://compositionstudy.com

Galvane, Q., Christie, M., Lino, C., & Ronfard, R. (2015). Camera-on-rails: Automated computation of constrained camera paths. In *Proceedings of the 8th acm siggraph conference on motion in games* (pp. 151–157). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/2822013.2822025 doi: 10.1145/2822013.2822025

Galvane, Q., Christie, M., Ronfard, R., Lim, C.-K., & Cani, M.-P. (2013). Steering behaviors for autonomous cameras. In *Proceedings of motion on games* (pp. 71:93–71:102). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/2522628.2522899 doi: 10.1145/2522628.2522899

Galvane, Q., Fleureau, J., Tariolle, F. L., & Guillotel, P. (2016). Automated cinematography with unmanned aerial vehicles. In *Proceedings of the eurographics workshop on intelligent cinematography and editing* (pp. 23–30). Goslar Germany, Germany: Eurographics Association. Retrieved from https://doi.org/10.2312/wiced.20161097 doi: 10.2312/wiced.20161097

Galvane, Q., Lino, C., Christie, M., Fleureau, J., Servant, F., Tariolle, F.-l., & Guillotel, P. (2018, July). Directing cinematographic drones. *ACM Trans. Graph.*, *37*(3), 34:1–34:18. Retrieved from http://doi.acm.org/10.1145/3181975 doi: 10.1145/3181975

Galvane, Q., Ronfard, R., Christie, M., & Szilas, N. (2014). Narrative-driven camera control for cinematic replay of computer games. In *Proceedings of the seventh international conference on motion in games* (pp. 109–117). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/2668064.2668104 doi: 10.1145/2668064.2668104

Gandhi, D., Pinto, L., & Gupta, A. (2017, Sep.). Learning to fly by crashing. In *2017 ieee/rsj international conference on intelligent robots and systems (iros)* (p. 3948-3955). doi: 10.1109/IROS.2017.8206247

Gebhardt, C., Hepp, B., Nägeli, T., Stevšić, S., & Hilliges, O. (2016). Airways: Optimization-based planning of quadrotor trajectories according to high-level user goals. In *Proceedings of the 2016 chi conference on human factors in computing systems* (pp. 2508–2519). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/2858036.2858353 doi: 10.1145/2858036.2858353

Gebhardt, C., Stevšić, S., & Hilliges, O. (2018, July). Optimizing for aesthetically pleasing quadrotor camera motion. *ACM Trans. Graph.*, *37*(4), 90:1–90:11. Retrieved from http://doi.acm.org/10.1145/3197517.3201390 doi: 10.1145/3197517.3201390

Gooch, B., Reinhard, E., Moulding, C., & Shirley, P. (2001). Artistic composition for image creation. In S. J. Gortler & K. Myszkowski (Eds.), *Rendering techniques 2001* (pp. 83–88). Vienna: Springer Vienna.

Grill, T., & Scanlon, M. (1990). *Photographic composition*. Orlando, FL: American Photographic Book Publishing.

Gross, L. (2016). *Multi-touch through-the-lens drone control*. Unpublished master's thesis, Massachusetts Institute of Technology.

Gu, S., Holly, E., Lillicrap, T., & Levine, S. (2017, May). Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates. In *2017 ieee international conference on robotics and automation (icra)* (p. 3389-3396). doi: 10.1109/ICRA.2017.7989385

Gude, O. (2004). Postmodern principles: In search of a 21st century art education. *Art Education*, *57*(1), 6–14.

Haber, J. (2015). *Enhancing the functional design of a multi-touch uav ground control station*. Unpublished master's thesis, Ryerson University.

Haber, J., & Chung, J. (2016). Assessment of uav operator workload in a reconfigurable multi-touch ground control station environment. *Journal of Unmanned Vehicle Systems*, *4*(3), 203-216. Retrieved from https://doi.org/10.1139/juvs-2015-0039 doi: 10.1139/juvs-2015-0039

Hansen, J. P., Alapetite, A., MacKenzie, I. S., & Møllenbach, E. (2014). The use of gaze to control drones. In *Proceedings of the symposium on eye tracking research and applications* (pp. 27–34). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/2578153.2578156 doi: 10.1145/2578153.2578156

Hanson, A. J., & Wernert, E. A. (1997, Oct). Constrained 3d navigation with 2d controllers. In *Proceedings. visualization '97 (cat. no. 97cb36155)* (p. 175-182). doi: 10.1109/VISUAL.1997.663876

Hart, S. G. (2006). Nasa-task load index (nasa-tlx); 20 years later. In (Vol. 50, p. 904-908). Retrieved from https://doi.org/10.1177/154193120605000909 doi: 10.1177/154193120605000909

Hart, S. G., & Staveland, L. E. (1988). Development of nasa-tlx (task load index): Results of empirical and theoretical research. *Advances in psychology*, *52*, 139–183.

Hasselt, H. v., Guez, A., & Silver, D. (2016). Deep reinforcement learning with double q-learning. In *Proceedings of the thirtieth aaai conference on artificial intelligence* (pp. 2094–2100). AAAI Press. Retrieved from http://dl.acm.org/citation.cfm?id=3016100.3016191

He, K., Zhang, X., Ren, S., & Sun, J. (2016, June). Deep residual learning for image recognition. In *2016 ieee conference on computer vision and pattern recognition (cvpr)* (p. 770-778). doi: 10.1109/CVPR.2016.90

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735–1780.

Hou, Q., Cheng, M., Hu, X., Borji, A., Tu, Z., & Torr, P. H. S. (2019, April). Deeply supervised salient object detection with short connections. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *41*(4), 815-828. doi: 10.1109/TPAMI.2018.2815688

Huang, C., Gao, F., Pan, J., Yang, Z., Qiu, W., Chen, P., ... Cheng, K. T. (2018, May). Act: An autonomous drone cinematography system for action scenes. In *2018 ieee international conference on robotics and automation (icra)* (p. 7039-7046). doi: 10.1109/ICRA.2018.8460703

Huang, C., Lin, C.-E., Yang, Z., Kong, Y., Chen, P., Yang, X., & Cheng, K.-T. (2019). Learning to film from professional human motion videos. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 4244–4253).

Huang, C., Yang, Z., Kong, Y., Chen, P., Yang, X., & Cheng, K. T. (2018, Oct). Through-the-lens drone filming. In *2018 ieee/rsj international conference on intelligent robots and systems (iros)* (p. 4692-4699). doi: 10.1109/IROS.2018.8594333

Huang, C., Yang, Z., Kong, Y., Chen, P., Yang, X., & Cheng, K.-T. T. (2019). Learning to capture a film-look video with a camera drone. In *Robotics and automation, 2019. icra 2019. proceedings of the 2019 ieee international conference on.*

Jakobsen, O., & Johnson, E. (n.d.). Control architecture for a uav-mounted pan/tilt/roll camera gimbal. In *Infotech@aerospace.* Retrieved from `https://arc.aiaa.org/doi/abs/10.2514/6.2005-7145` doi: 10.2514/6.2005-7145

Jankowski, J., Hulin, T., & Hachet, M. (2014, March). A study of street-level navigation techniques in 3d digital cities on mobile touch devices. In *2014 ieee symposium on 3d user interfaces (3dui)* (p. 35-38). doi: 10.1109/3DUI.2014.6798838

Johnson-Roberson, M., Barto, C., Mehta, R., Sridhar, S. N., Rosaen, K., & Vasudevan, R. (2017, May). Driving in the matrix: Can virtual worlds replace human-generated annotations for real world tasks? In *2017 ieee international conference on robotics and automation (icra)* (p. 746-753). doi: 10.1109/ICRA.2017.7989092

Joubert, N., Roberts, M., Truong, A., Berthouzoz, F., & Hanrahan, P. (2015, October). An interactive tool for designing quadrotor camera shots. *ACM Trans. Graph.*, *34*(6), 238:1–238:11. Retrieved from `http://doi.acm.org/10.1145/2816795.2818106` doi: 10.1145/2816795.2818106

Kahn, G., Villaflor, A., Pong, V., Abbeel, P., & Levine, S. (2017). Uncertainty-aware reinforcement learning for collision avoidance. *CoRR*, *abs/1702.01182*. Retrieved from `http://arxiv.org/abs/1702.01182`

Kang, H., Li, H., Zhang, J., Lu, X., & Benes, B. (2018, Oct). Flycam: Multitouch gesture controlled drone gimbal photography. *IEEE Robotics and Automation Letters*, *3*(4), 3717-3724. doi: 10.1109/LRA.2018.2856271

Kendall, A., Hawke, J., Janz, D., Mazur, P., Reda, D., Allen, J., ... Shah, A. (2018). Learning to drive in a day. *CoRR*, *abs/1807.00412*. Retrieved from http://arxiv.org/abs/1807.00412

Khan, A., Komalo, B., Stam, J., Fitzmaurice, G., & Kurtenbach, G. (2005). Hovercam: Interactive 3d navigation for proximal object inspection. In *Proceedings of the 2005 symposium on interactive 3d graphics and games* (pp. 73–80). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/1053427.1053439 doi: 10.1145/1053427.1053439

Kong, S., Shen, X., Lin, Z., Mech, R., & Fowlkes, C. (2016). Photo aesthetics ranking network with attributes and content adaptation. In B. Leibe, J. Matas, N. Sebe, & M. Welling (Eds.), *Computer vision – eccv 2016* (pp. 662–679). Cham: Springer International Publishing.

Krahenbuhl, P. (2018, June). Free supervision from video games. In *2018 ieee/cvf conference on computer vision and pattern recognition* (p. 2955-2964). doi: 10.1109/CVPR.2018.00312

KUa, C.-J., & Chen, L.-C. (2014). A study on the natural manipulation of multi-touch gestures for 3d object rotation using a large touch screen. In *Universal design 2014: Three days of creativity and diversity: Proc. ud* (Vol. 35, p. 279). doi: 10.3233/978-1-61499-403-9-279

LaFleur, K., Cassady, K., Doud, A., Shades, K., Rogin, E., & He, B. (2013, jun). Quadcopter control in three-dimensional space using a noninvasive motor imagery-based brain–computer interface. *Journal of Neural Engineering*, *10*(4), 046003. Retrieved from https://doi.org/10.1088 doi: 10.1088/1741-2560/10/4/046003

Lan, Z., Shridhar, M., Hsu, D., & Zhao, S. (2017, July). Xpose: Reinventing user interaction with flying cameras. In *Proc. rss.* Cambridge, Massachusetts. doi: 10.15607/RSS.2017.XIII.006

Landau, M., & van Delden, S. (2017). A system architecture for hands-free uav drone control using intuitive voice commands. In *Proceedings of the companion of the 2017 acm/ieee international conference on human-robot interaction* (pp. 181–182). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/3029798.3038329 doi: 10.1145/3029798.3038329

Lichtenstern, M., Frassl, M., Perun, B., & Angermann, M. (2012, March). A prototyping environment for interaction between a human and a robotic multi-agent system. In *2012 7th acm/ieee international conference on human-robot interaction (hri)* (p. 185-186). doi: 10.1145/2157689.2157747

Lino, C., & Christie, M. (2012). Efficient composition for virtual camera control. In *Proceedings of the acm siggraph/eurographics symposium on computer animation* (pp. 65–70). Goslar Germany, Germany: Eurographics Association. Retrieved from http://dl.acm.org/citation.cfm?id=2422356.2422367

Lino, C., & Christie, M. (2015, July). Intuitive and efficient camera control with the toric space. *ACM Trans. Graph.*, *34*(4), 82:1–82:12. Retrieved from http://doi.acm.org/10.1145/2766965 doi: 10.1145/2766965

Liu, L., Chen, R., Wolf, L., & Cohen-Or, D. (2010). Optimizing photo composition. *Computer Graphics Forum*, *29*(2), 469-478. Retrieved from https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-8659.2009.01616.x doi: 10.1111/j.1467-8659.2009.01616.x

Lu, H., Li, Y., Mu, S., Wang, D., Kim, H., & Serikawa, S. (2018, Aug). Motor anomaly detection for unmanned aerial vehicles using reinforcement learning. *IEEE Internet of Things Journal*, *5*(4), 2315-2322. doi: 10.1109/JIOT.2017.2737479

Mendes, D., Sousa, M., Ferreira, A., & Jorge, J. (2014). Thumbcam: Returning to single touch interactions to explore 3d virtual environments. In *Proceedings of the ninth acm international conference on interactive tabletops and surfaces* (pp. 403–408). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/2669485.2669554 doi: 10.1145/2669485.2669554

Micire, M., Drury, J. L., Keyes, B., & Yanco, H. A. (2009). Multi-touch interaction for robot control. In *Proceedings of the 14th international conference on intelligent user interfaces* (pp. 425–428). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/1502650.1502712 doi: 10.1145/1502650.1502712

Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T., Harley, T., ... Kavukcuoglu, K. (2016, 20–22 Jun). Asynchronous methods for deep reinforcement learning. In M. F. Balcan & K. Q. Weinberger (Eds.), *Proceedings of the 33rd international conference on machine learning* (Vol. 48, pp. 1928–1937). New York, New York, USA: PMLR. Retrieved from http://proceedings.mlr.press/v48/mniha16.html

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., ... others (2015). Human-level control through deep reinforcement learning. *Nature*, *518*(7540), 529. doi: 10.1038/nature14236

Mo, K., Li, H., Lin, Z., & Lee, J. (2018). The adobeindoornav dataset: Towards deep reinforcement learning based real-world indoor robot visual navigation. *CoRR*, *abs/1802.08824*. Retrieved from http://arxiv.org/abs/1802.08824

Monajjemi, V. M., Wawerla, J., Vaughan, R., & Mori, G. (2013, Nov). Hri in the sky: Creating and commanding teams of uavs with a vision-mediated gestural interface. In *2013 ieee/rsj international conference on intelligent robots and systems* (p. 617-623). doi: 10.1109/IROS.2013.6696415

Mueller, M., Casser, V., Lahoud, J., Smith, N., & Ghanem, B. (2017). Ue4sim: A photo-realistic simulator for computer vision applications. *CoRR*, *abs/1708.05869*. Retrieved from http://arxiv.org/abs/1708.05869

Murray, N., Marchesotti, L., & Perronnin, F. (2012, June). Ava: A large-scale database for aesthetic visual analysis. In *2012 ieee conference on computer vision and pattern recognition* (p. 2408-2415). doi: 10.1109/CVPR.2012.6247954

Myung-Jin Kim, Song, T., Jin, S., Jung, S., Gi-Hoon Go, Kwon, K., & Jeon, J. (2010, Oct). Automatically available photographer robot for controlling composition and taking pictures. In *2010 ieee/rsj international conference on intelligent robots and systems* (p. 6010-6015). doi: 10.1109/IROS.2010.5650341

Nägeli, T., Meier, L., Domahidi, A., Alonso-Mora, J., & Hilliges, O. (2017, July). Real-time planning for automated multi-view drone cinematography. *ACM Trans. Graph.*, *36*(4), 132:1–132:10. Retrieved from `http://doi.acm.org/10.1145/3072959.3073712` doi: 10.1145/3072959.3073712

Nagi, J., Giusti, A., Caro, G. A. D., & Gambardella, L. M. (2014, March). Human control of uavs using face pose estimates and hand gestures. In *2014 9th acm/ieee international conference on human-robot interaction (hri)* (p. 1-2).

Nagi, J., Giusti, A., Gambardella, L. M., & Di Caro, G. A. (2014, Sep.). Human-swarm interaction using spatial gestures. In *2014 ieee/rsj international conference on intelligent robots and systems* (p. 3834-3841). doi: 10.1109/IROS.2014.6943101

Neff, A. E., Lee, D., Chitrakaran, V. K., Dawson, D. M., & Burg, T. C. (2007, March). Velocity control for a quad-rotor uav fly-by-camera interface. In *Proceedings 2007 ieee southeastcon* (p. 273-278). doi: 10.1109/SECON.2007.342901

Ng, W. S., & Sharlin, E. (2011, July). Collocated interaction with flying robots. In *2011 ro-man* (p. 143-149). doi: 10.1109/ROMAN.2011.6005280

Ngeli, T., Alonso-Mora, J., Domahidi, A., Rus, D., & Hilliges, O. (2017, July). Real-time motion planning for aerial videography with dynamic obstacle avoidance and viewpoint optimization. *IEEE Robotics and Automation Letters*, *2*(3), 1696-1703. doi: 10.1109/LRA.2017.2665693

Obaid, M., Kistler, F., Kasparavičiūtė, G., Yantaç, A. E., & Fjeld, M. (2016). How would you gesture navigate a drone?: A user-centered approach to control a drone. In *Proceedings of the 20th international academic mindtrek conference* (pp. 113–121). New York, NY, USA: ACM. Retrieved from `http://doi.acm.org/10.1145/2994310.2994348` doi: 10.1145/2994310.2994348

Opeyemi, A., Prior, S., T Thomas, G., Saddington, P., & Ramchurn, S. (2019, 01). Multimodal human aerobotic interaction. In (p. 142-165). doi: 10.4018/978-1-5225-8365-3.ch006

Ortega, F. R. (2014). *3d navigation with six degrees-of-freedom using a multi-touch display*. Unpublished doctoral dissertation, Florida International University.

Paravati, G., Sanna, A., Lamberti, F., & Celozzi, C. (2011, Sep.). A reconfigurable multi-touch framework for teleoperation tasks. In *Etfa2011* (p. 1-4). doi: 10.1109/ETFA.2011.6059219

Parrot. (2012, April). Parrot ar.drone 2.0 user guide [Computer software manual]. (`http://ardrone2.parrot.com`)

Peshkova, E., Hitz, M., & Ahlström, D. (2017). Exploring user-defined gestures and voice commands to control an unmanned aerial vehicle. In R. Poppe, J.-J. Meyer, R. Veltkamp, & M. Dastani (Eds.), *Intelligent technologies for interactive entertainment* (pp. 47–62). Cham: Springer International Publishing.

Peshkova, E., Hitz, M., & Kaufmann, B. (2017, Jan). Natural interaction techniques for an unmanned aerial vehicle system. *IEEE Pervasive Computing*, *16*(1), 34-42. doi: 10.1109/MPRV.2017.3

Pfeil, K., Koh, S. L., & LaViola, J. (2013). Exploring 3d gesture metaphors for interaction with unmanned aerial vehicles. In *Proceedings of the 2013 international conference on intelligent user interfaces* (pp. 257–266). New York, NY, USA: ACM. Retrieved from `http://doi.acm.org/10.1145/2449396.2449429` doi: 10.1145/2449396.2449429

Qiu, W., Zhong, F., Zhang, Y., Qiao, S., Xiao, Z., Kim, T. S., & Wang, Y. (2017). Unrealcv: Virtual worlds for computer vision. In *Proceedings of the 25th acm international conference on multimedia* (pp. 1221–1224). New York, NY, USA: ACM. Retrieved from `http://doi.acm.org/10.1145/3123266.3129396` doi: 10.1145/3123266.3129396

Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., ... Ng, A. Y. (2009). Ros: an open-source robot operating system. In *Icra workshop on open source software* (Vol. 3, p. 5).

Quigley, M., Goodrich, M. A., Griffiths, S., Eldredge, A., & Beard, R. W. (2005, April). Target acquisition, localization, and surveillance using a fixed-wing mini-uav and gimbaled camera. In *Proceedings of the 2005 ieee international conference on robotics and automation* (p. 2600-2605). doi: 10.1109/ROBOT.2005.1570505

Richter, S. R., Hayder, Z., & Koltun, V. (2017, Oct). Playing for benchmarks. In *2017 ieee international conference on computer vision (iccv)* (p. 2232-2241). doi: 10.1109/ICCV.2017.243

Roberts, M., & Hanrahan, P. (2016, July). Generating dynamically feasible trajectories for quadrotor cameras. *ACM Trans. Graph.*, *35*(4), 61:1–61:11. Retrieved from `http://doi.acm.org/10.1145/2897824.2925980` doi: 10.1145/2897824.2925980

Sadeghi, F., & Levine, S. (2016). (cad)$ˆ2$rl: Real single-image flight without a single real image. *CoRR*, *abs/1611.04201*. Retrieved from `http://arxiv.org/abs/1611.04201`

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. (2018, June). Mobilenetv2: Inverted residuals and linear bottlenecks. In *2018 ieee/cvf conference on computer vision and pattern recognition* (p. 4510-4520). doi: 10.1109/CVPR.2018.00474

Sandru, L. A., Crainic, M. F., Savu, D., Moldovan, C., Dolga, V., & Preitl, S. (2016). Automatic control of a quadcopter, ar. drone, using a smart glove. In *Proceedings of the 4th international conference on control, mechatronics and automation* (pp. 92–98). New York, NY, USA: ACM. Retrieved from `http://doi.acm.org/10.1145/3029610.3029619` doi: 10.1145/3029610.3029619

Sanna, A., Lamberti, F., Paravati, G., Henao Ramirez, E. A., & Manuri, F. (2012). A kinect-based natural interface for quadrotor control. In A. Camurri & C. Costa (Eds.), *Intelligent technologies for interactive entertainment* (pp. 48–56). Berlin, Heidelberg: Springer Berlin Heidelberg.

Sarkar, A., Patel, K. A., Ram, R. K. G., & Capoor, G. K. (2016, March). Gesture control of drone using a motion controller. In *2016 international conference on industrial informatics and computer systems (ciics)* (p. 1-5). doi: 10.1109/ICCSII.2016.7462401

Schulman, J., Levine, S., Abbeel, P., Jordan, M., & Moritz, P. (2015, 07–09 Jul). Trust region policy optimization. In F. Bach & D. Blei (Eds.), *Proceedings of the 32nd international conference on machine learning* (Vol. 37, pp. 1889–1897). Lille, France: PMLR. Retrieved from `http://proceedings.mlr.press/v37/schulman15.html`

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). Proximal policy optimization algorithms. *CoRR*, *abs/1707.06347*. Retrieved from `http://arxiv.org/abs/1707.06347`

Shah, S., Dey, D., Lovett, C., & Kapoor, A. (2018). Airsim: High-fidelity visual and physical simulation for autonomous vehicles. In M. Hutter & R. Siegwart (Eds.), *Field and service robotics* (pp. 621–635). Cham: Springer International Publishing.

Silver, D. (2016, June). *Deep reinforcement learning, a tutorial at icml 2016.* Retrieved from `http://icml.cc/2016/tutorials/deep_rl_tutorial.pdf.pdf`

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., ... others (2016). Mastering the game of go with deep neural networks and tree search. *nature*, *529*(7587), 484. doi: 10.1038/nature16961

Sugiura, Y., Fernando, C. L., Withana, A. I., Kakehi, G., Sakamoto, D., Sugimoto, M., ... Inakage, M. (2009). An operating method for a bipedal walking robot for entertainment. In *Acm siggraph asia 2009 art gallery & emerging technologies: Adaptation* (pp. 79–79). New York, NY, USA: ACM. Retrieved from `http://doi.acm.org/10.1145/1665137.1665198` doi: 10.1145/1665137.1665198

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction.* MIT press.

Tai, L., Paolo, G., & Liu, M. (2017, Sep.). Virtual-to-real deep reinforcement learning: Continuous control of mobile robots for mapless navigation. In *2017 ieee/rsj international conference on intelligent robots and systems (iros)* (p. 31-36). doi: 10.1109/IROS.2017.8202134

Teixeira, J. M., Ferreira, R., Santos, M., & Teichrieb, V. (2014, May). Teleoperation using google glass and ar, drone for structural inspection. In *2014 xvi symposium on virtual and augmented reality* (p. 28-36). doi: 10.1109/SVR.2014.42

Trujillo, A. C., Puig-Navarro, J., Mehdi, S. B., & McQuarry, A. K. (2017). Using natural language to enable mission managers to control multiple heterogeneous uavs. In P. Savage-Knepshield & J. Chen (Eds.), *Advances in human factors in robots and unmanned systems* (pp. 267–280). Cham: Springer International Publishing.

Wei, Z., Zhang, J., Shen, X., Lin, Z., Mech, R., Hoai, M., & Samaras, D. (2018, June). Good view hunting: Learning photo composition from dense view pairs. In *2018 ieee/cvf conference on computer vision and pattern recognition* (p. 5437-5446). doi: 10.1109/CVPR.2018.00570

Won, J., Park, J., & Lee, J. (2018, December). Aerobatics control of flying creatures via self-regulated learning. *ACM Trans. Graph.*, *37*(6), 181:1–181:10. Retrieved from `http://doi.acm.org/10.1145/3272127.3275023` doi: 10.1145/3272127.3275023

Wu, C. (2011). *Visualsfm: A visual structure from motion system.* Retrieved from `http://ccwu.me/vsfm/`

Wu, C., Agarwal, S., Curless, B., & Seitz, S. M. (2011, June). Multicore bundle adjustment. In *Cvpr 2011* (p. 3057-3064). doi: 10.1109/CVPR.2011.5995552

Xie, K., Yang, H., Huang, S., Lischinski, D., Christie, M., Xu, K., ... Huang, H. (2018, July). Creating and chaining camera moves for qadrotor videography. *ACM Trans. Graph.*, *37*(4), 88:1–88:13. Retrieved from `http://doi.acm.org/10.1145/3197517.3201284` doi: 10.1145/3197517.3201284

Yu, A. J., & Cohen, J. D. (2008). Sequential effects: Superstition or rational behavior? *Advances in neural information processing systems*, *21*, 1873-1880. Retrieved from `https://www.ncbi.nlm.nih.gov/pubmed/26412953`

Yu, Y., He, D., Hua, W., Li, S., Qi, Y., Wang, Y., & Pan, G. (2012). Flyingbuddy2: A brain-controlled assistant for the handicapped. In *Proceedings of the 2012 acm conference on ubiquitous computing* (pp. 669–670). New York, NY, USA: ACM. Retrieved from `http://doi.acm.org/10.1145/2370216.2370359` doi: 10.1145/2370216.2370359

Yuneec. (2016a, August). Skyview user manual v1.0 [Computer software manual]. (`http://us.yuneec.com/skyview-goggles`)

Yuneec. (2016b, September). Typhoon h user manual rs v1.2 [Computer software manual]. (`http://us.yuneec.com/typhoon-h-overview`)

Zabarauskas, M., & Cameron, S. (2014, May). Luke: An autonomous robot photographer. In *2014 ieee international conference on robotics and automation (icra)* (p. 1809-1815). doi: 10.1109/ICRA.2014.6907096

Zhang, F., Leitner, J., Milford, M., Upcroft, B., & Corke, P. I. (2015). Towards vision-based deep reinforcement learning for robotic motion control. *CoRR*, *abs/1511.03791*. Retrieved from `http://arxiv.org/abs/1511.03791`

Zhu, Y., Mottaghi, R., Kolve, E., Lim, J. J., Gupta, A., Fei-Fei, L., & Farhadi, A. (2017, May). Target-driven visual navigation in indoor scenes using deep reinforcement learning. In *2017 ieee international conference on robotics and automation (icra)* (p. 3357-3364). doi: 10.1109/ICRA.2017.7989381

VITA

VITA

Hao Kang was born in Dandong, Liaoning, China in 1987. He received his B.E. degree in software engineering from Beijing Jiaotong University, Beijing, China, in 2000 and M.S. degree in game design and simulation and Engineer's degree in computer science from Stevens Institute of Technology, Hoboken, NJ, in 2013 and 2015. He will receive the Ph.D. degree in computer graphics technology from Purdue University, West Lafayette, IN, in 2019.

He was a Research Assistant with Purdue High Performance Computer Graphics Laboratory from 2015 to 2019, under the guidance of Dr. Bedrich Benes. He was a Course Instructor of Purdue CGT 110-Technical Graphics Communication from 2015 to 2017. He was awarded Purdue Polytechnic Institute Graduate Teaching Assistant Scholarship in the year 2016-2017 for his teaching activity. He worked as a Research Intern with Siemens Healthineers in 2017 summer and Adobe Research in 2018 spring. His research interests include software engineering, human-computer interaction, 3D computer graphics and vision, machine learning and robotics.

Publications

Kang, H., Fiser, M., Shi, B., Sheibani, F., Hirst, P., & Benes, B. (2016, Nov). IMapple: functional structural model of apple trees. In 2016 ieee international conference on functional-structural plant growth modeling, simulation, visualization and applications (fspma) (p. 90-97). doi: 10.1109/FSPMA.2016.7818293

Kang, H., Li, H., Zhang, J., Lu, X., & Benes, B. (2018, Oct). FlyCam: Multitouch gesture controlled drone gimbal photography. IEEE Robotics and Automation Letters,3(4),3717-3724. doi: 10.1109/LRA.2018.2856271

Pirk, S., Krs, V., Hu, K., Rajasekaran, S. D., Kang, H., Yoshiyasu, Y., . . . Guibas, L. J.(2017, June). Understanding and exploiting object interaction landscapes. ACM Trans. Graph.,36(3). doi: 10.1145/3083725