UAV, Come To Me: End-to-End, Multi-Scale Situated HRI with an Uninstrumented Human and a Distant UAV

Mani Monajjemi, Sepehr Mohaimenianpour, and Richard Vaughan*

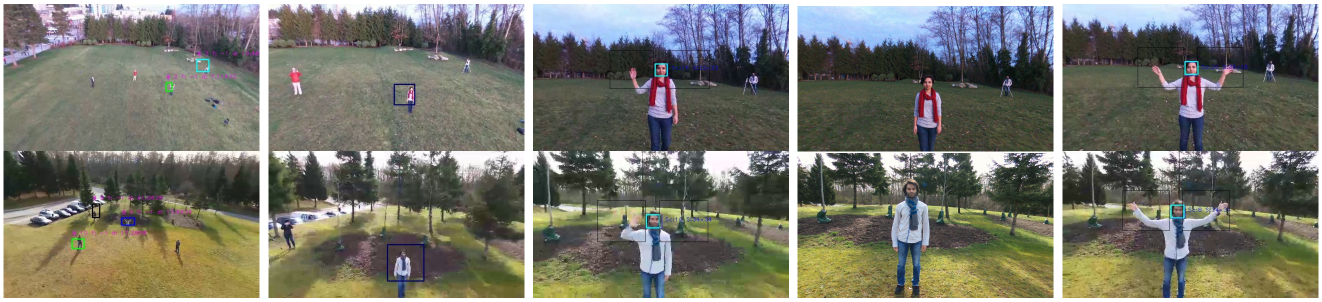

Fig. 1: Our system in action during two of outdoor experiments (Section IV-B). The flying robot's view from left to right: (i) The user initiates the interaction with the UAV using a dual-arm waving gesture in the presence of other humans (distance is $\approx 30m$) (ii) The UAV approaches the user using an appearance based tracker and a custom cascade controller (iii) The user asks the UAV to take a picture of her using a single hand waving gesture (iv) The resulting portrait (v) The user terminates the interaction by performing a dual hand waving gesture.

***Abstract*— We present the first demonstration of end-to-end far-to-near situated interaction between an uninstrumented human user and an initially distant outdoor autonomous Unmanned Aerial Vehicle (UAV). The user uses an arm-waving gesture as a signal to attract the UAV's attention from a distance. Once this signal is detected, the UAV approaches the user using appearance-based tracking until it is close enough to detect the human's face. Once in this close-range interaction setting, the user is able to use hand gestures to communicate its commands to the UAV. Throughout the interaction, the UAV uses colored-light-based feedback to communicate its intent to the user. We developed this system to work reliably with a low-cost consumer UAV, with only computation off-board. We describe each component of this interaction system, giving details of the depth estimation strategy and the cascade predictive flight controller for approaching the user. We also present experimental results on the performance of the complete system and its individual components.**

I. Introduction

The rapid development of low-cost Unmanned Air Vehicles (UAVs) is enabling many valuable applications and new industries are growing around them. In particular small multi-rotor vehicles are relatively safe to operate around humans, so we have recently been able to consider situated and embodied human-UAV interactions as an alternative to conventional remote-control systems. We have previously argued that hands-free, embodied, sensor-mediated interaction could be useful in some UAV applications [1], [2].

In this paper we show the first realized Human-Robot Interaction system whereby an uninstrumented user can attract the attention of a distant (20 to 30 meters) autonomous outdoor flying robot, the robot then approaches the user to close range ($\approx$ 2 meters), hovers facing the user, then responds appropriately to a small vocabulary of hand gestures.

The main contributions of the paper are (i) the first demonstration of end-to-end interaction with a distant flying robot over multiple scales (ii) a description of a robust integrated visual servo and predictive cascade controller design for smooth approach towards a human and (iii) a case study in outdoor situated HRI with UAVs over multiple scales. Below we briefly survey work in situated interaction with UAVs, then describe the components of our end-to-end situated interaction system. We describe how we use fast computer vision methods to detect the user's intention from distance using a monocular camera, how we estimate depth when approaching the user, a predictive cascade controller to follow a smooth trajectory towards the user despite the high latency of our off-board vision via WiFi link, our close-range interaction system for the communication of commands from the user to the UAV and our colored-light-based feedback system for communicating the UAV's state to the user. We present experimental results of this system in action, where an uninstrumented user can summon a Parrot Bebop Drone from distances over $20m$ and have the robot take a close range portrait photo - a *selfie* - of her. The scale change is such that the person initially appears around 15 pixels high in the UAV's 640×368 camera image, but the portrait taken features the person's torso and head in the center of the image (Figure 1).

II. Background

We previously presented systems that enable uninstrumented humans to perform close range situated interaction

* Autonomy Lab, School of Computing Science, Simon Fraser University {mmonajje, smohaime, vaughan}@sfu.ca

with UAVs through gaze and hand gestures [1], [3], [4] and obtain a distant UAV's attention using stationary periodic gestures while the UAV is in flight [2]. In this work, we build upon those systems to provide an end-to-end interaction system for human-flying robot interaction. Motivated by Wilderness Search And Rescue (WiSAR) and personal filming drone application domains, we identify the components of an end-to-end interaction system as (i) explicit interaction initiation; (ii) approach and re-positioning to facilitate close-range interaction; (iii) communication of commands and intents from the human to the UAV; and (iv) communication of intents from the UAV to the human.

Uninstrumented interaction initiation between co-located humans and UAVs mostly happens in two forms. In the first form, the UAV utilizes vision-based human feature detectors to find potential interaction partners. Alternatively the user may try to attract the UAV's attention by using active stimuli such as gestures, sound or body movements.

Using vision-based human detectors on-board a UAV poses multiple challenges. First, when the UAV is flying far from the humans, features are either hard to detect or require high computational resources to be detected in real-time. Some researchers use extra sensors such as thermal cameras [5], scene information such as saliency maps [6] or the prior on the height of the human combined with ground plane estimation [7] to identify regions of interest in the image plane before executing vision-based human detection.

Most existing human detectors assume an upright human view [5]. The violation of this assumption caused by time-varying and different vantage point of UAVs causes the second issue for performing on-board pedestrian detection. In [6] the authors show that the performance of a conventional pedestrian detector can be improved by retraining it using a dataset that is recorded from a UAV and with synthetic variations of camera roll and pitch angles. In [8], the authors propose to compensate for this time-varying vantage point by estimating the ground plane using UAV's telemetry data and cancel out the distortion by projecting the image to the ground plane prior to using a pedestrian detector. Although none of the aforementioned methods were explicitly used for human-UAV interaction, they are applicable for implicit interaction initiation or as a building block for explicit interaction initiation. The same is true for methods such as [9] that utilize moving object detection to find regions of interest and potential interaction partners in the UAV's Field Of View (FOV).

Once the interaction between a human and a UAV (or a team of UAVs) is initiated, the human and the UAV(s) can interact more directly by communicating their intents. Uninstrumented, natural and situated communication of commands from humans to UAVs have been recently explored by researchers in form of human studies and practical systems. Example human studies include [10] and [11] that investigate natural commanding modalities for collocated interaction between a human and a flock or a single UAV respectively.

To approach towards the user, the UAV should first track the location of the user, then constantly control its flight trajectory to reach the person. Recently, researchers have applied state of the art long-term appearance-based visual trackers and Image Based Visual Servo (IBVS) control for following an uninstrumented human with a UAV [12], [13]. We use the same long-term visual tracker developed by [13] in our system. Similar to [12], we use a visual servo controller to generate approach trajectories for our target platform. However, since our system performs approaching towards the user, rather than following her, depth estimation of the target becomes more critical, thus we provide a solution to estimate depth of the tracked object using UAV's telemetry data and the intrinsic parameters of the camera. Furthermore, unlike [12], our system is not initialized by a human operator, instead it uses explicit interaction signals from the human to initialize the appearance-based tracker. Most relevant to our work is [14] in which the authors designed a self-contained person follower UAV that implements implicit interaction initiation through pedestrian detection, appearance based tracking, depth estimation and trajectory controller. As mentioned, we use explicit interaction initiation signals that helps the UAV steer its attention to a single person when multiple users are in its FOV. In addition, we explicitly address the two-way communication of intents and commands between a human and a UAV.

Practical systems for situated interaction with UAVs in the literature mainly utilize sound and gestural interfaces for communication of commands from humans to UAVs. In the prototype environment of [15], the authors use a Microsoft Kinect sensor on-board a hovering UAV to transmit gestural commands to a team of flying robots in an indoor environment. In [16], the authors propose a solution for canceling the ego-motion of an RGB-D camera attached to a flying UAV and use the stabilized depth image to perform gesture recognition and person following in an indoor environment. In [17], the authors applied transfer learning to develop a person-specific gestural interface to command a UAV.

As argued in [18] being able to "talk" is as important requirement as being able to "listen" for an autonomous agent. Through proper feedback, the user can understand if the UAV correctly understands her intents and if the UAV is functioning properly. These in turn decrease the user's cognitive workload and improve her awareness and safety. Recently, a few different modalities for communication of intent and affects from a UAV to its collocated human partners have been studied. These modalities include flight path manipulation [19], [20] and light-based feedback systems [21]. To the best of our knowledge, we are the first to demonstrate an end-to-end human-flying robot interaction system that implements all these components in outdoor settings and bring a robot from relatively long distances to a proximate distance to the user.

III. METHOD

Our proposed system consists of three hardware components and five major software blocks. We use the Parrot Bebop, a lightweight quad-rotor consumer UAV as our

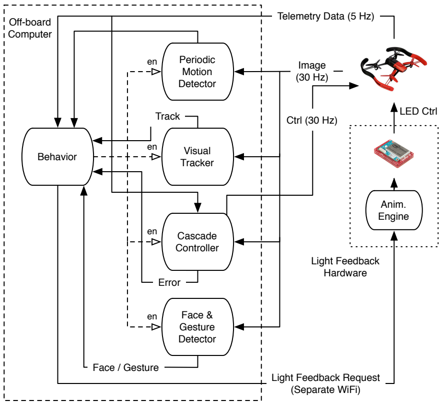

Fig. 2: The block diagram of the system

platform. The UAV transmits the live video stream of its front facing camera and flight telemetry data to an off-board computer over WiFi. This computer runs the core software components of the interaction system and sends the desired control commands over the same WiFi link to the UAV. A small form factor computer is mounted on top of the UAV to drive an array of 11 high intensity RGB LEDs mounted on the front side of the UAV and generates feedback signals (Figure 3). The off-board computer communicates the desired feedback to the embedded computer over a separate WiFi link based on the current state of the interaction. The five major components of the software stack are the behavior generator and coordinator, the long-range periodic motion detector for initiating the interaction, the appearance-based object tracker, the cascade controller used for approach towards the user, the face engagement detector and motion based gesture recognizer for the close-range interaction phase. Figure 2 shows the overall architecture of our interaction system.

The system starts in the *searching* state, where it looks for periodic but net-stationary motions in camera's FOV. When a periodic signal is detected, the corresponding region of the image is fed into a long-term visual tracker which simultaneously tracks the object in the image plane and refines its appearance model. The track is piped into a cascade controller, which first estimates the distance of the target with respect to the image plane, then controls the flight of the UAV towards the target. The approach towards the target ends either when the target is in the center of the image plane and the UAV is within a pre-defined distance with respect to the target, or a human face detector finds a human face inside the target's bounding box in the image plane. In the latter case, the system transitions into *close-range interaction* state, where the UAV maintains the user's face in the center of its FOV and at a fixed distance from its camera (using the same cascade controller). In this state, a motion based gesture detector detects the left hand and right hand waving gesture of the human which is consequently used to command the vehicle to perform a certain action. As mentioned earlier, the UAV constantly communicates its state and intentions to the user using its front facing colored-light-based feedback system. In the remainder of this section we describe each component of this system in more details.

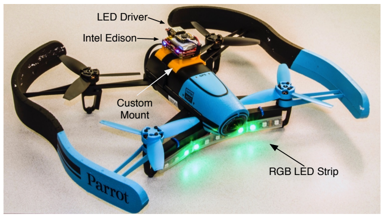

Fig. 3: Our platform. Parrot Bebop Drone and color-light-based feedback system. The UAV is executing the *Gaze* feedback (Section III-F).

A. Hardware Platform

One of the difficulties faced during the development of this system was to choose a UAV platform suitable for close-range situated interaction with a human. The main criteria for this platform were safety around the interaction partner, being able to perform stable hovering and carrying enough payload for sensing, feedback and computation. Most consumer UAVs available in the market nowadays are multi-rotor flying platforms that are able to perform stable hovering. Many of these platforms are also powerful enough to carry small form factor sensing devices and computational units. However, not many of these UAVs provide the minimum safety measures to fly in close proximity of people. We believe any UAV platform that enters the *social space* ($\approx 3-4m$) [22] of a human or closer should be at least equipped with physical propeller guards and provide an automatic shutdown systems in case of contact between any of its propellers and an object.

We chose the *Parrot Bebop Drone*[1] as our UAV platform. Although this UAV provides the required minimum safety measures, its on-board flight controller computer is not powerful enough to execute our CPU intensive software stack. Due to its limited payload carrying capabilities, it is also not capable of carrying powerful computing devices. For these reasons we opted to control the UAV off-board over WiFi. Bebop is a lightweight consumer quad-rotor UAV with an on-board high definition camera and a Fisheye lens with the FOV of 180 degrees. The video stream of this camera is digitally stabilized and rectified on-board prior to being transmitted over WiFi with the reduced resolution of $640px \times 368px$ at 30 frames per second. The rectification target is limited to the FOV of $\approx 80^\circ$ (horizontal) and $\approx 50^\circ$ (vertical), essentially simulating a virtual pan/tilt camera with a stabilized gimbal. The desired pan and tilt of this camera is also controllable over WiFi. Bebop transmits its telemetry data (i.e. altitude and attitude) over WiFi to the off-board computer at the rate of 5 Hz.

B. Interaction Initiation using Periodic Gestures

To initiate the interaction with a distant human and while the UAV is in flight, we use the system previously developed to detect periodic salient motions on-board a UAV [2]. The

[1] http://www.parrot.com/products/bebop-drone/

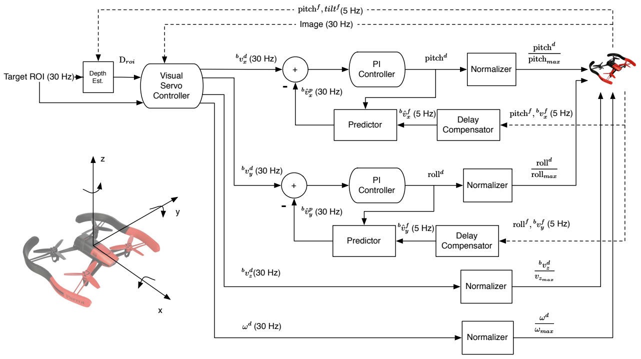
$\text{pitch}^f, tilt^f$(5 Hz)
Image (30 Hz)
Target ROI (30 Hz)
Depth Est.
D_{roi}
Visual Servo Controller
${}^b v_x^d$ (30 Hz)
+
-
${}^b \hat{v}_x^p$ (30 Hz)
PI Controller
pitch^d
Normalizer
$\frac{\text{pitch}^d}{\text{pitch}_{max}}$
Predictor
${}^b \hat{v}_x^f$ (5 Hz)
Delay Compensator
$\text{pitch}^f, {}^b v_x^f$ (5 Hz)
${}^b v_y^d$ (30 Hz)
+
-
${}^b \hat{v}_y^p$ (30 Hz)
PI Controller
roll^d
Normalizer
$\frac{\text{roll}^d}{\text{roll}_{max}}$
Predictor
${}^b \hat{v}_y^f$ (5 Hz)
Delay Compensator
$\text{roll}^f, {}^b v_y^f$ (5 Hz)
${}^b v_z^d$(30 Hz)
Normalizer
$\frac{{}^b v_z^d}{v_{z_{max}}}$
ω^d (30 Hz)
Normalizer
$\frac{\omega^d}{\omega_{max}}$
z
y
x

Fig. 4: The block diagram of the cascade controller.

main software component of this system (Freely available at `http://autonomylab.org/obzerver/`) is a real-time computer vision pipeline that detects salient moving objects that exhibit periodic motion patterns in a moving camera's FOV. The dual-arm waving of a human is a periodic signal (with a dominant frequency of 1 to 4 Hz) which is detected by this component to initiate the interaction. In [2] we provided a detailed description of this computer vision pipeline.

C. *Visual Tracker*

To track the location of the target detected by interaction initiation module in the image plane, we use the long-term visual tracker of [13]. This appearance-based tracker combines correlation filters [23] for short-term tracking with tracking-learning-detection (TLD) framework [24] for long-term tracking, target re-detection and loss detection.

D. *Cascade Controller for Approaching the User*

The task of the approach controller is to bring the UAV to a pre-defined distance of the user while keeping her in the center of its FOV. We designed a cascade controller in order to achieve this task. The input to the cascade controller is the current location of the tracked object in the image plane and the outputs are the desired set-point velocities for the on-board flight controller of the UAV. As a quad-rotor UAV, the Bebop has four controllable Degrees Of Freedom (DOF): *roll*, *pitch*, *yaw* and *altitude*. The on-board flight controller of Bebop offers velocity control for the latter two DOF. However, *roll* and *pitch* - which control the acceleration of the UAV in lateral and forward directions - are set directly. The Bebop performs on-board visual-inertial state estimation and reports the estimated values for its attitude and velocity at 5 Hz. The high level controller in this cascade is an IBVS controller that receives the current position of the tracked Region Of Interest (ROI) in the camera plane, estimates its depth based on the current state of the UAV, then calculates a set of reference velocities that would bring the camera to the desired location in front of the user. The angular and vertical velocity components of the IBVS controller's output are sent directly to the UAV, while the lateral and forward velocity components are fed into a velocity controller which deals with the latency and slow update rate of the feedback signal.

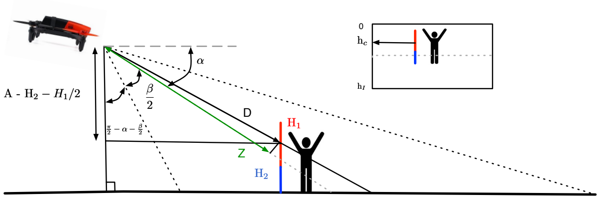

Fig. 5: The proposed method for estimating the depth of the tracked object. Refer to the text (Section III-D.1) for details.

Figure 4 shows the architecture of the approach controller. Internally this controller uses the dynamic model of the UAV to compensate the delay and predict the feedback signal as well as a PI controller to track the reference velocity. We provide more details about this controller in the following sections.

1) Depth Estimation: Similar to the approach proposed in [14], we use camera intrinsic parameters (specifically β, its vertical FOV), the prior on the size of the tracked object (H_1), the prior on the distance of the object from the ground plane (H_2), current tilt of the camera with respect to the inertial frame of the UAV (α), and the vertical pixel location of the ROI in the image plane (h_c) to estimate the distance of the center of the object (Z) from the image plane, under the assumptions that the ground plane is flat (horizontal) and the user's ROI is perpendicular to the ground plane (Figure 5). Using simple geometry we can derive Z as:

$$Z = \frac{A - H_2 - \frac{H_1}{2}}{sin(\frac{\pi}{2} - \alpha - \frac{\beta}{2})} \cdot cos(\frac{h_c - \frac{h_I}{2}}{h_I} \cdot \beta) \qquad (1)$$

2) Visual Servo Control: Once the depth of the tracked bounding box is estimated, we use a classical IBVS controller [25] to calculate the desired velocity of the camera to approach the target. We use the four corners of a rectangle formed by projecting a rectangular target with the size of $H \times W$ and desired depth of Z^d to the camera plane as the desired visual features of the IBVS. We consider the four corners of the tracked bounding box in the image plane as observations and augment these feature points with the estimated depth of the center of the bounding box (Z). These feature points represent a rectangle parallel to the image plane and provide only three independent constraints to the IBVS controller to calculate the desired velocity of the camera (${}^c\mathbf{v}$). This implies that one DOF of the UAV is not controllable. We chose the lateral movement of the UAV (${}^b v_y$) as the non-controllable DOF and map ${}^c\mathbf{v}$ to Bebop's DOFs as follows. We transform ${}^c v_x$ and ${}^c v_z$ to the forward (${}^b v_x$) and vertical (${}^b v_z$) DOFs through the tilt angle of the camera ($[{}^b v_x, {}^b v_z]^T = \Re(\alpha)[{}^c v_z, {}^c v_x]$), set (${}^b v_y$) to 0 and control the angular velocity (ω) directly from the error between the horizontal center of the bounding box and the horizontal center of the image plane.

This control schema flies the UAV to a semi-sphere with radius Z^d in front of the target in a configuration that keeps the object at the center of its FOV. If α is 0, the altitude of the UAV at the end of the approach trajectory will be $H_2 + H_1/2$.

Although the semi-spherical shape of the final location with respect to the target might not be suitable for applications such as perching or landing on a moving platform, for HRI applications it is not a major concern since the user can reposition herself (changes her yaw or gaze direction) towards the UAV when the robot is flying towards her.

3) Velocity Controller: As mentioned in Section III-D, ${}^b v_z$ and ω are directly sent to the on-board flight controller of the Bebop for execution. For lateral and forward velocities, we designed a velocity controller to control the roll and pitch angles of the UAV such that it tracks the desired velocity vector. Our objective was to design a controller that generates a smooth trajectory towards the user. The major challenges towards designing such a controller are the latency and low update rate of the feedback signal. Our proposed controller uses a dynamic model of the UAV to compensate for this latency and predict the feedback signal. The dynamic model we used is a first-order non-linear system that relates the roll and pitch angles of the Bebop to its lateral and forward velocities respectively (Equation 2).

$$\begin{aligned} \dot{v}_x^b &= C_x v_x^b + g\ tan(pitch) \\ \dot{v}_y^b &= C_y v_y^b - g\ tan(roll) \end{aligned} \tag{2}$$

In this equation, g is the gravitational constant ($\approx 9.81 s^{-2}$) and C_x and C_y are the free parameters. We performed a system identification step to find C_x and C_y by flying the UAV indoors and measuring true values for $pitch$, $roll$, ${}^b v_x$ and ${}^b v_y$ using a high precision and frequency ($\approx$ 120 Hz) motion capture system. The estimated values for these parameters are $C_x = 0.576\ s^{-1}$ and $C_y = 0.585\ s^{-1}$. By minimizing the squared error between measured velocities and feedback velocities over different time offsets, we estimated the latency of the feedback as $t_d \approx 262$ milliseconds. This latency is mainly caused by the WiFi transport delay as well as the down-sampling/buffering step performed by Bebop's firmware prior to sending the feedback over WiFi.

As shown in Figure 4, once the feedback is received, the controller utilizes the dynamic model of the UAV to predict the state of the system (${}^b \hat{v}_x$ and ${}^b \hat{v}_y$) from the feedback signals (${}^b v_x^f$, ${}^b v_y^f$, roll^f and pitch^f) which are t_d seconds delayed. The PI controllers calculate the desired control values of the system (pitch^d and roll^d) by calculating the error between the feedback velocity and the desired velocity (coming from the visual servo controller). Instead of relying on the low-frequency feedback to generate the control signal which would either decrease the output rate to 5 Hz or increase its jerk because of the periodically increasing delay between the last feedback signal and the true state of the system, the controller again utilizes the dynamic model of the UAV to predict the state of the system from the last received feedback signal and the latest desired control command. Once a new feedback signal is received, it resets the state of the predictor. This way, the predictor fills the 200 milliseconds gap between two feedback readings to provide a 30 Hz estimation of this signal for the PI controller.

Feedback Animation	State	Metaphor
Search	Searching	Radar Scanner
Approach	Approaching	Pointing
Engaged	Close-range	Gaze
Selfie	Close-range	Camera Timer
Bye	Close-range	Iris
Bad Video	Any	-
Lost	Approach & Close-range	Radar Scanner

TABLE I: Animations used for providing light-based feedback to the user, their corresponding state and metaphors.

E. Close-range Interaction

When the UAV enters the *Approaching State*, the behavior coordinator enables the close-range interaction component of the software stack. This component consists of a human face detector and an optical flow based gesture detector. We previously used this component for close-range interaction with a group of flying robots [1]. The UAV uses the cascade classifier of Viola and Jones [26] to detect human faces in the image plane. It only considers the faces which their corresponding bounding boxes overlap with the tracked object's region in the image. It also uses the so called *face score* [27] to filter out the faces that the classifier is not confident about. Once a candidate face is detected, it is internally tracked with a Kalman filter and its bounding box is continuously fed into the cascade controller, replacing the input from the visual tracker. Compared to the output of the tracker, the tracked bounding box of the face region is more consistent with the prior on its size. Therefor, in case the face is detected, the resulting depth estimation will be more accurate which subsequently leads to a more precise positioning in front of the user. The UAV maintains its position on a semi-sphere around the user while keeping her face in the center of its FOV.

While tracking the face, the close-range interaction component calculates the dense optical flow inside two regions around the human face. The size of these regions are linearly dependent on the size of the face and are placed such that they capture hand/arm movements. In order to cancel out the effect of ego-motion of the UAV, the median of magnitude of optical flow vectors inside the human face and the background regions are subtracted from all optical flow vectors inside the two gesture regions. A post processing step smooths the time variations of average flow per pixel inside the gesture regions, then applies a median filter and thresholding to decide if there is substantial motion in any of those regions. These motions are considered as left/right hand waving gestures by this component and used to communicate commands from the human to the UAV.

F. Communication of Intents from the UAV to the User

To communicate the state of the UAV and its intents to the user, we developed a custom color-light-based feedback system. This feedback systems consists of 11 individually addressable RGB LEDs mounted on the front side of the UAV, a AVR-based driver board and an Intel Edison embedded computer that executes the feedback generation software (Figure 3). The high level behavior coordinator (which runs on the off-board computer), communicates over WiFi to this embedded computer to request the execution of any of predefined animations based on the current state of the UAV and its next command. A custom key-frame-based animation engine runs on-board the Edison computer to generate the feedback signals. The total weight overhead of this feedback system is 55 grams.

As shown in [21], using light feedback helps co-located humans deduce the flying intent of a UAV faster and more accurately. We believe this feedback modality is advantageous to other modalities previously used in this context such as sound [3] over long distances, specifically for small form factor UAVs. In [21], the authors showed that animations based on *Gaze* and *car blinker* metaphors perform well to communicate the flying intention of a UAV. Inspired by these results, we designed a set of feedback signals to communicate the intent of the UAV to the user during each phase of the interaction process. These designed signals use colors and motion to convey the intent to the user. Table I provides a summary for all these feedback signals and their corresponding metaphors. Please refer to the supplementary video for the visualization of these signals.

IV. Experiments

For all the experiments we used the platform described in Section III-A. Except for the LED animation generator, all the software ran on a notebook computer with a specification matching the small form factor embedded computer we previously used for self-contained Human-UAV interaction[2] [2]. For the data intensive communication with the UAV over WiFi, we used a long-range IEEE 802.11ac external network card with a high gain antenna. We extensively used ROS [28] to integrate different software and hardware components of this system. The cascade controller internally uses the ViSP library [29] to perform IBVS. More details about our platform, as well as the source code for various components of this system (including the details about the light-feedback hardware, the ROS driver for Parrot Bebop Drone, ROS bindings for the long-term visual tracker, the cascade controller and the animation generator engine) is available at `autonomylab.org/bebop_hri/`.

A. Approach Controller

The goal of this experiment was to validate the approach controller and assess its depth estimation accuracy as well as the resulting approach trajectories. We performed this experiment in a $7m \times 11m \times 3m$ indoor environment, equipped with a Vicon motion capture system. We put an augmented reality marker of size 56×56 centimeters in a fixed location of the arena (marked with X in Figures 6). The height of the center of the target from the ground was $1.175m$. The augmented reality marker was used to bootstrap the long-range interaction initiation part of the system and to replace the visual tracker for one leg of the experiments, therefor we did not use the 6 DOF localization data these markers provide. Instead, we use the axis-aligned bounding box of detected marker in the image plane to initiate or replace the tracker.

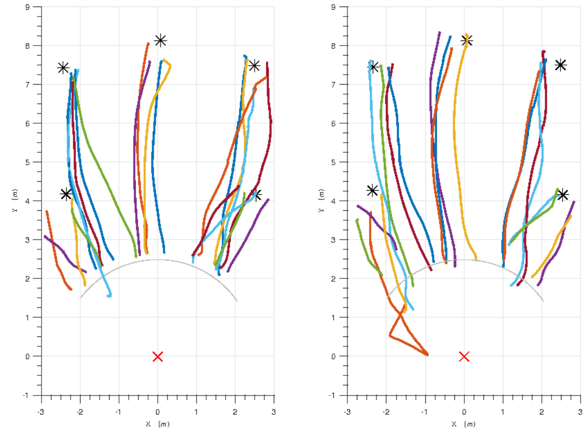

Fig. 6: The indoor approach trajectories for leg 1 [continuous detection] (left) and leg 2 [detect, then track] (right)

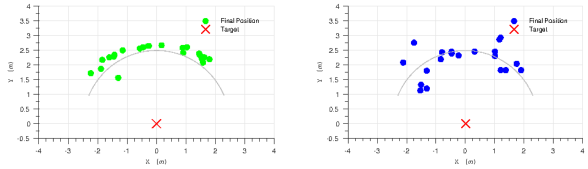

Fig. 7: The indoor approach location error for leg 1 [continuous detection] (left) and leg 2 [detect, then track] (right)

In the first part of the experiment, the UAV was placed in one of 5 pre-defined starting locations in the room (Marked with $\star$ in Figure 6), either looking towards the target or looking forward (aligned with y-axis of the room). After takeoff, when the UAV first detects the marker, it transitions to the approaching mode and constantly uses the consequent detections to feed the approach controller, replacing the visual tracker. The desired depth of the UAV with respect to the target and the camera tilt was set to $2.5m$ and $0°$ respectively. Once the sum of the velocity errors were below a certain threshold, the UAV would land. The second leg of the experiment was similar in design with the first leg. The only difference was that the marker detection was only used once to initiate the visual tracker which would provide the reference bounding box to the approach controller. We repeated each leg of this experiment four times from each location (two for each orientation), resulting in total of 20 experiments for each leg.

Figure 7 shows the 2D top-down view of the ground-truth locations where the UAV decided to land relative to the target for both legs of the experiments. Since the camera was not tilted during these experiments ($\alpha = 0$, the target altitude of the UAV is expected to converge to the aforementioned height of the target center ($1.175m$). The root mean squared (RMS) error of distance and altitude error of the UAV for the first leg of the experiment was $0.242m$ and $0.064m$ respectively. The same errors measured for the second leg of the experiment (with the visual tracker in the loop) were $0.392m$ and $0.076m$. The RMS depth estimation error for

[2] Intel 5th generation Core i5 CPU, 8GB of RAM, SSD Storage

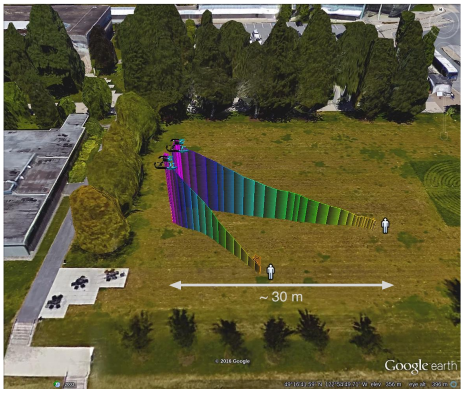

Fig. 8: 3D rendering of two outdoor approach trajectories from actual GPS log data

the two legs of the experiment were $0.665m$ and $0.793m$ respectively. Figure 6 shows the trajectories the UAV flew to reach the target in 2D for each leg of the experiment.

Since in leg 1 the detection happens in every frame, the input to the approach controller more accurately corresponds to the true location of the target in the image plane, therefore the depth estimation accuracy is higher and the final location error is less for the first leg of the experiments. This means when an object detector is used to drive the approach controller or when the tracker does not drift much, and when the prior on the object size is precise, the final location of the UAV with respect to the target is more accurate.

B. Outdoor Experiments

To demonstrate and validate the proposed end-to-end Human-UAV interaction system, we performed a series of outdoor experiments with the platform and the setup previously described in Sections III-A and IV. The tilt angle of the virtual camera was set to $45°$ and this value was dynamically and independently being controlled by the behavior coordinator to smoothly tilt it to $0°$ towards the end of approach trajectory. We tested the interaction system in three different locations, at three different times of the day (noon, early and late afternoon) and with 9 users. All the users were from our own research group, but not necessarily familiar with the details of the interaction system in advance. In each experiment, the UAV would take off from a fixed location and towards a pre-defined direction. A safety pilot would correct the direction of the UAV after take-off to cancel out the yaw error during takeoff, then would put the UAV in autonomous mode. The UAV's search behavior was to hover at the fixed altitude of $12m$ and tilt its camera down to $45°$. During each experiment, one user would try to attract the UAV's attention by using dual arm waving gestures. The other user(s) would act as distractors either by walking or standing in the FOV of the UAV. The UAV would execute the behavior described in Section III-A to find its interaction partner, approach and engage in close-range interaction with her, while constantly provide light-based feedback as described in Section III-F. In close-range interaction mode, the single hand waving gesture of the user would cause the UAV to take a close-range portrait photo - a *selfie* - of her. The user then would ask the UAV to terminate the interaction and leave by performing a double hand waving gesture (*Bye Bye*). Upon receiving this command, the UAV would turn away, ascend and restart its behavior from the *searching state*. We briefed each user once in advance about the interpretation of each feedback signal. For the outdoor experiments we set the prior on the size of the region of the periodic motion (H_1) to $1.5m$ and the prior on its distance from the ground (H_2) to $1m$.

State	**Attempts**	**Success Count (%)**
Search	52	42 (80%)
Approach	42	40 (95%)
Close-range tracking	40	38 (95%)
Selfie Gesture	40	38 (95%)
Terminate Gesture	38	37 (97%)
Feedback System	52	51 (98%)
Total	**52**	**37** (71%)

TABLE II: The summary of failures in the end-to-end outdoor experiments

We consider an experiment to be end-to-end successful when the human and the UAV perform all steps of the interaction scenario. The incidents that would fail an experiment were: search behavior does not succeed in less than 45 seconds or detects a false positive, the approach behavior loses the target over the course of the approach and does not recover in 30 seconds, the UAV does not detect the user's face before getting closer than $0.5m$ to her, and when any of gestural commands fail after more than one retries. Table II provides a summary of the failure points for all the 52 experiments. Taking all these failure points into account, 37 out of 52 experiments (71%) were successful end-to-end. Figure 1 shows snapshots from the UAV's FOV during each phase of the interaction process, except for high resolution *selfie* shots, all other images are the actual image inputs to our system. Figure 8 shows two sample 3D approach trajectories generated from GPS readings of the UAV.

As the breakdown in Table II shows, the major failure point of the system was in the search behavior for interaction initiation (periodic motion detector). Other components of the system performed with $\geq 95\%$ reliability. We also observed that for a few experiments the search behavior took a relatively long time to find the person of interest. We measured the average and standard deviation of the response time of this component for successful runs as 34.82 and 17.02 seconds, respectively. From the 10 failures, 6 of them were due to false positives and 4 were due to the timeout (false negatives). We further analyzed the failure cases of this component by looking into the effect of different conditions on the failure. We observed that the variable frame-rate of the input video stream, the low contrast between the user and the background, the MPEG artifacts due to variable bit-rate control were among the most affecting factors. The low contrast between the user and the background were mainly caused by sunlight, failures in automatic white-balancing of the UAV's camera and the similarity of the color of user's clothes to the background which makes the user a less salient object in the environment. An immediate direction for future work is to improve this component's performance in realworld settings and decrease its response time. Similar to indoor trajectories (Figure 6), the outdoor flight trajectories of the UAV were smooth, and were able to steer the UAV towards the user at the maximum speed of $\approx 2.5\ ms^{-1}$. The appearance based tracker performed well with occasional positional and scale drift. However, since upon detecting a face, the system would re-estimate its depth, those drifts did not cause major failures for the *approaching* behavior. We can informally report that the close-range interaction system was responsive and users found the color-light-based feedback system informative and intuitive. For future work we are planning to formally assess the intuitiveness and usability of this system by performing human user studies.

V. Conclusion And Future Work

In this paper we presented the first demonstration of end-to-end human-UAV interaction in outdoor environments that implements (i) explicit interaction initiation; (ii) approach and re-positioning towards the user; (iii) close-range communication of commands from the user to the UAV; and (iv) communication of intents from the UAV to the user. We show how the user can use dual arm-waving gesture to attract a flying robot's attention from distance, how an integrated visual tracking and servoing system can bring the robot to the close proximity of the user and how the user can perform close-range interaction with the UAV after the approach. Effective velocity control of the UAV based on computer vision was achieved despite a high latency control loop. We also describe how the UAV employs color-light-based feedback to keep the human informed about its intents. We implemented this system on a low-cost consumer UAV that we believe meets the minimum safety requirements for the close-range interaction with a human. In a series of indoor and outdoor experiments we validated our integrated system, analyzed the accuracy of our depth estimation and approach trajectories and identified major failure points of the system. Future work includes making the interaction initiation more robust and responsive and performing formal user studies on the usability and intuitiveness of the end-to-end system and its individual components. Designing a unified visual servo controller to simultaneously control the pan and tilt angle of the camera as the UAV approaches the user is another possible future research direction.

References

[1] V. M. Monajjemi, J. Wawerla, R. Vaughan, and G. Mori, "HRI in The Sky: Creating and commanding teams of UAVs with a vision-mediated gestural interface," in *Int. Conf. on Intelligent Robots and Systems (IROS)*, 2013.

[2] V. M. Monajjemi, J. Bruce, S. A. Sadat, J. Wawerla, and R. Vaughan, "UAV, do you see me? Establishing mutual attention between an uninstrumented human and an outdoor UAV in flight," in *Int. Conf. on Intelligent Robots and Systems (IROS)*, 2015.

[3] S. Pourmehr, V. M. Monajjemi, S. A. Sadat, F. Zhan, J. Wawerla, G. Mori, and R. Vaughan, ""You are green:" A touch-to-name interaction in an integrated multi-modal multi-robot HRI system," in *IEEE/ACM Int. Conf. on HRI*, 2014.

[4] V. M. Monajjemi, S. Pourmehr, S. A. Sadat, F. Zhan, J. Wawerla, G. Mori, and R. Vaughan, "Integrating Multi-modal Interfaces to Command UAVs," in *ACM Int. Conf. on HRI*, 2014.

[5] P. Blondel, A. Potelle, C. Pegard, and R. Lozano, "Fast and viewpoint robust human detection for SAR operations," in *IEEE Int. Symp. on Safety, Security, and Rescue Robotics (SSRR)*, 2014.

[6] ——, "Human detection in uncluttered environments: From ground to UAV view," *Int. Conf. on Control Automation Robotics & Vision*, 2014.

[7] F. De Smedt, D. Hulens, and T. Goedeme, "On-board real-time tracking of pedestrians on a UAV," in *Embedded Vision Workshop (CVPR)*, 2015.

[8] M. Andriluka, P. Schnitzspan, J. Meyer, S. Kohlbrecher, K. Petersen, O. Von Stryk, S. Roth, and B. Schiele, "Vision based victim detection from unmanned aerial vehicles," *Int. Conf. on Intelligent Robots and Systems (IROS)*, 2010.

[9] B. Jung and G. S. Sukhatme, "Real-time Motion Tracking from a Mobile Robot," *Int. J. of Social Robotics*, vol. 2, pp. 63–78, 2009.

[10] G. Jones, N. Berthouze, R. Bielski, and S. Julier, "Towards a situated, multimodal interface for multiple UAV control," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2010.

[11] J. R. Cauchard, J. L. E, K. Y. Zhai, and J. A. Landay, "Drone & Me: An Exploration into natural human-drone interaction," in *Int. Joint Conf. on Pervasive and Ubiquitous Computing (UbiComp)*, 2015.

[12] J. Pestana, J. L. Sánchez-López, S. Saripalli, and P. Campoy, "Computer vision based general object following for GPS-denied multirotor unmanned vehicles," in *American Control Conf. (ACC)*. IEEE, 2014.

[13] K. Haag, S. Dotenco, and F. Gallwitz, "Correlation filter based visual trackers for person pursuit using a low-cost quadrotor," in *IEEE Int. Conf. on Innovations for Community Services (I4CS)*, 2015.

[14] M. Danelljan, F. S. Khan, M. Felsberg, K. Granström, F. Heintz, P. Rudol, M. Wzorek, J. Kvarnström, and P. Doherty, "A low-level active vision framework for collaborative unmanned aircraft systems," in *Advances in Autonomous Robotics Systems*. Springer, 2015.

[15] M. Lichtenstern, M. Frassl, B. Perun, and M. Angermann, "A prototyping environment for interaction between a human and a robotic multi-agent system," in *ACM Int. Conf. on HRI*, 2012.

[16] T. Naseer, J. Sturm, and D. Cremers, "FollowMe: Person following and gesture recognition with a quadrocopter," in *Int. Conf. on Intelligent Robots and Systems (IROS)*, 2013.

[17] G. Costante, E. Bellocchio, P. Valigi, and E. Ricci, "Personalizing vision-based gestural interfaces for HRI with UAVs: a transfer learning approach," in *Int. Conf. on Inteligent Robots and Systems*, 2014.

[18] H. Jones and S. Rock, "Dialogue-based human-robot interaction for space construction teams," in *IEEE Aerospace Conf.*, 2002.

[19] M. Sharma, D. Hildebrandt, G. Newman, J. E. Young, and R. Eskicioglu, "Communicating affect via flight path Exploring use of the Laban Effort System for designing affective locomotion paths," in *ACM/IEEE Int. Conf. on HRI*, 2013.

[20] D. Szafir, B. Mutlu, and T. Fong, "Communication of intent in assistive free flyers," in *ACM/IEEE Int. Conf. on HRI*, 2014.

[21] ——, "Communicating directionality in flying robots," in *ACM/IEEE Int. Conf. on HRI*, 2015.

[22] E. T. Hall, *The hidden dimension*. Doubleday, 1966.

[23] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *Pattern Analysis and Machine Intelligence (PAMI)*, 2015.

[24] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-Learning-Detection," *Pattern Analysis and Machine Intelligence (PAMI)*, 2012.

[25] F. Chaumette and S. Hutchinson, "Visual servo control. I. Basic approaches," *Robotics & Automation*, vol. 13, no. 4, pp. 82–90, 2006.

[26] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Computer Vision and Pattern Recognition*, 2001.

[27] A. Couture-Beil, R. T. Vaughan, and G. Mori, "Selecting and commanding individual robots in a multi-robot system," in *Canadian Conf. on Computer and Robot Vision (CRV)*, 2010.

[28] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: an open-source Robot Operating System," in *ICRA workshop on open source software*, 2009.

[29] E. Marchand, F. Spindler, and F. Chaumette, "ViSP for visual servoing: a generic software platform with a wide class of robot control skills," *Robotics Automation Magazine*, vol. 12, no. 4, pp. 40–52, 2005.