Moving target tracking method for unmanned aerial vehicle/unmanned ground vehicle heterogeneous system based on AprilTags

Xiao Liang¹,², Guodong Chen¹,², Shirou Zhao¹ and Yiwei Xiu¹

Abstract
Using the characteristics of unmanned aerial vehicle/unmanned ground vehicle, heterogeneous systems can accomplish many complex tasks cooperatively. Moving target tracking is an important basis for the relative positioning and formation maintenance of heterogeneous cooperative systems. This paper first introduces the unmanned aerial vehicle/unmanned ground vehicle collaborative tracking task and heterogeneous system. In order to maintain the original stability of unmanned aerial vehicle, a control method based on SBUS protocol to simulate remote control is proposed. About unmanned ground vehicle with Mecanum wheel, a detailed description of control method is designed. For the problems of real-time performance and occlusion, a tracking scheme based on AprilTag identification is studied. The scheme tracks the Tag target in the case of no occlusion. When occlusion occurs, the scheme tracks the color feature around the Tag. The accuracy of the tracking algorithm and the problem of occlusion are greatly improved. Finally, the scheme is applied to the heterogeneous systems. Simulation and experimental results show that the proposed method is suitable for unmanned aerial vehicle/unmanned ground vehicle heterogeneous system to perform the collaborative tracking task.

Keywords
Unmanned aerial vehicle/unmanned ground vehicle collaborative system, moving target tracking, analog remote control, Mecanum unmanned ground vehicle, AprilTag identification

Date received: 9 May 2019; accepted: 28 October 2019

Introduction
With the rapid development of science and technology, the applications of aerial robots (also known as unmanned aerial vehicles (UAVs)) and ground robots (unmanned ground vehicles (UGVs)) are receiving more and more attention from researchers all over the world. Recently, UAV/UGV heterogeneous system¹–³ has become a hot topic. In many missions, UAVs can quickly scout vast areas. On the contrary, UGVs can accurately locate ground targets and complete complex interactions, but move slowly. Therefore, UAV and UGV can cooperate to finish many complex tasks.

In UAV/UGV heterogeneous system, insufficient real-time performance and accuracy will lead to failures in collaboration. Therefore, the target tracking algorithm is an important basis for UAV/UGV to make relative positioning and formation.⁴–⁶ The task of the target tracking is to establish the positional relationship of object in a continuous video sequence. Since the attitude, scale, occlusion, and light are constantly changing during the movement, many researches have been carried out.

Moving target recognition and tracking mainly include the following classic methods: frame difference method,⁷ MeanShift algorithm,⁸ and optical flow method.⁹ The method of frame difference calculation is fast and usually used in camera calibration. MeanShift is simple and easy to implement. However, the tracking performance is not good for targets that move faster or change in scale. CamShift¹⁰–¹² was improved by the MeanShift to solve the scale problem. CamShift converts the image from RGB space to HSV space, to
reduce the effect of illumination. Since only the color feature is used, the tracking performance is poor when the color of background is complex or close to the color of the target. The optical flow method is easily affected by illumination. To find the target again after losing the target, a combination of detection and tracking is proposed, such as the TLD (tracking-learning-detection) method. \(^13\) TLD algorithm achieves long-term tracking of single targets and solves the problem of losing target which is caused by target deformation and partial occlusion. However, in global search, the real-time performance is not satisfactory. The correlation filtering algorithm adopts the local search method which has obvious advantages in real-time performance, but it is difficult to track the target with high speed and longer occlusion time, such as KCF (kernel correlation filter)\(^14\) based on HOG (histogram of oriented gradients) feature. In 2016, Danelljan et al. \(^15\) proposed the ECO (Efficient Convolution Operators) target tracking algorithm after C-COT (Learning Continuous Convolution Operators for Visual Tracking), \(^16\) which greatly improved the target tracking speed. The ECO deep learning version can reach 8 FPS/s, and the ECO-HC can reach 60 FPS/s, which is a better tracking algorithm. However, heterogeneous system requires a high level of accuracy and real-time tracking, so they are not suitable for installation on embedded devices.

Many scholars apply tracking algorithms to unmanned devices. Qu et al. \(^17\) studied long-term reliable visual tracking of UAVs. They compared KCF and TLD with their proposed algorithm. However, UAV is only used to shoot videos as a test dataset for comparison and the research has not been verified by physical experiments. AprilTag\(^18,19\) is a visual reference library that is widely used in UAV positioning guidance. Xiao et al. \(^20\) used AprilTag to visually locate tethered UAV, but their experiment was carried out in a barrier-free indoor environment. Wang et al. \(^21\) used AprilTags to implement UAV tracking of UGV, but the method was only tested in Gazebo-sim simulation software.

This paper first introduces a UAV/UGV heterogeneous system. Then a quadrotor control method based on SBUS protocol to simulate remote control (RC) is proposed. The design and control of UGV with Mecanum wheel are also introduced. To improve the real-time performance and accuracy, a tracking scheme based on AprilTag is proposed and color information is added around the original Tag in the case of occlusion. In experiment, the tracking scheme is loaded into the performed UAV/UGV heterogeneous systems and the result shows that the proposed method is suitable for collaborative tracking task. The experiment consists of two parts: first, a simulation experiment is used to verify the feasibility of the AprilTag algorithm and, second, a physical experiment is completed by the heterogeneous system composed of the quadrotor and the Maltese unmanned vehicle. The main contribution of this paper is the establishment and verification of the system that can provide a physical experiment platform for different control and image processing methods.

### System structure

The heterogeneous system proposed in this paper is composed of UAV and UGV. The prototype is developed with quadrotor as UAV and Mecanum vehicle as UGV, and the system is tested in many physical experiments at last. Therefore, “UAV/UGV heterogeneous system” introduces the structure of UAV/UGV heterogeneous system with quadrotor and Mecanum vehicle as examples, and section “UAV/UGV tracking system” introduces the collaborative tracking function of the heterogeneous system.

#### UAV/UGV heterogeneous system

In the air, UAV with high speed can detect vast areas and make aerial fire suppression, but it is limited in the accuracy of localization on the ground. On the contrary, UGV has the disadvantage of not being able to move rapidly or see through obstacles, but it can carry out complex and accurate interaction with environment. The heterogeneous system consisting of UAV and UGV is not simply changed from “single agent” to “multiple agents.” The heterogeneous feature will bring unique advantages to collaborative mission and make \(1 + 1 > 2\).

A UAV equipped with camera and other sensors can obtain a two-dimensional (2D) horizontal image of the environment in front of UGV. It supplements the obstacle information in front of UGV, so it can provide local/global image information for UGV obstacle avoidance. Based on the mutual awareness of UAV/UGV, heterogeneous system can achieve complex tasks such as cluster formation, avoidance guidance, and information fusion (Figure 1).

#### UAV/UGV tracking system

In Figure 1, the gray block in the diagram is relative positioning and it is particularly critical for the heterogeneous system. Although GPS (Global Positioning System) can achieve relative positioning of two types of unmanned devices, the accuracy is low. Collaborative positioning becomes an important connection between UAV and UGV for the heterogeneous collaborative system to complete collaborative tasks. Since the collaborative system works relatively close, the tracking accuracy and real-time performance are the key issues that must be solved.

The tracking system of this paper consists of two parts: UAV tracking subsystem and UGV tracking subsystem (Figure 2). UGV automatically tracks front moving targets with AprilTag. UAV can automatically track UGV and get the relative position and attitude between UGV.
In the figure, the upper left is a UAV, which is equipped with a camera that recognizes the UGV below. In the middle is UGV, which has an AprilTag tag (tag36h11_1) on its deck. The UGV is equipped with a camera that recognizes moving targets in front. The lower right is a moving target with an AprilTag attached.

**Design and control of UAV**

The UAV used in the heterogeneous system is a quad-rotor. The main structure consists of a frame, four motors, four electronic governors, a signal translator, a flight controller, and wireless communication equipment. Accelerometer, gyroscope, magnetometer, and barometer on the flight controller can realize attitude solving. The structure of hardware is shown in Figure 3.

OpenMV camera can acquire the images in real time and send the processed coordinates to the UAV. The UAV automatically tracks the target according to the coordinates. In this system, RC has the highest control right and can be switched to manual mode at any time.

**UAV modeling and automatic control**

At present, there are many control algorithms for quad-rotor, such as linear quadratic regulator (LQR), adaptive control, genetic algorithm, proportional-integral-derivative (PID) controller, and so on. To
ensure the stability of control, PID method is adopted in the paper and later physical experiments.

First, the model of quadrotor is established. Assuming that the quadrotor is rigid and symmetrical, and the origin of the body frame is consistent with the center of gravity, the quadrotor dynamics model can be described as

\[
\begin{align*}
\dot{x} &= u_1 \left( \cos c \sin c + \sin c \cos c + d_x \right) \\
\dot{y} &= u_1 \left( \sin c \sin c - \cos c \sin c + d_y \right) \\
\dot{z} &= u_1 \cos c \cos c + g + d_z \\
\phi &= u_2 \\
\theta &= u_3 \\
\psi &= u_4
\end{align*}
\]

(1)

where \( \theta \) is the pitch angle, \( \psi \) is the yaw angle, \( \phi \) is the roll angle, \( g \) is the gravitational acceleration, and \( d_x \), \( d_y \), and \( d_z \) are the additional disturbances caused by the external force. The total thrust on the \( z \)-axis is expressed as \( u_1 = \sum_{i=1}^{4} F_i/m \), where \( m \) is the mass of the quadrotor and \( F_i (i = 1, 2, 3, 4) \) is the thrust of the four rotors. \( u_1 \) and \( u_1 \) are the pitch and roll inputs, respectively, and \( u_1 \) represents the yawing moment.

The structure of the PID controller system is shown in Figure 4.

**Analog RC based on SBUS protocol**

In this paper, the main contribution of UAV control is to propose a multi-rotor autonomous flight control method based on SBUS protocol analog RC. A signal generation device is used to simulate the generation of RC signals to control UAV autonomous flight. This method can maintain the original stability of UAV and reduce the workload of developing flight control. The method uses a signal generation device to simulate the RC signals to control UAV. A Pixhawk flight controller is used and connected to a signal translator (Figure 5). The function of signal translator is to decode and encode the input signal of RC and generate the control signal for flight controller, so as to realize automatic/semi-automatic control of the UAV.

The UAV has three flight modes: manual RC mode, automatic flight mode, and emergency stop mode. The flow chart of control is shown in Figure 6.

1. **Manual RC mode**
   The receiver receives RC data in real time and sends it to the signal translator through the SBUS protocol. The signal translator decodes and encodes the data and sends the received data to flight controller through SBUS protocol as usual, to achieve the purpose of manual control flight.

2. **Automatic flight mode**
   The signal translator decodes the data transmitted from the receiver, but does not perform the code transmission to flight controller. Instead, the controller calculates the signal of
each channel and then encodes and transmits these signals to achieve the purpose of automatic flight.

3. Emergency stop mode
   The signal translator immediately sets all channel values to the middle value and then encodes and sends them to achieve the purpose of UAV fixed-point hovering to ensure flight safety.

Target tracking of UAV

The image information under the UAV is collected by an OpenMV camera and determines whether there is a Tag. If there is a Tag, the deviation between the center of the tag and the center of the image is calculated. Then, the control system calculates the control amount of UAV through the incremental PID, encodes the data and sends it to UAV to track the target. If no target is detected or target is lost, UAV automatically hovers until it finds the target or receives landing command.

Design and control of UGV

The UGV is mainly composed of a car body, four motors and their drives, a main control board, and a wireless communication equipment (Figure 7).

Modeling and control of UGV

To simplify the mathematical model of kinematics, there are some assumptions:

1. The omnidirectional wheel will not slip, and the ground has sufficient friction;
2. Four wheels are distributed on four corners of the rectangle or square, and wheels are parallel to each other.

Assuming that the body coordinate system coincides with the geographic coordinate system, the Mecanum UGV motion direction is specified (Figure 8).

According to Figure 8, the motion of UGV can be linearly decomposed into three components: movement along X, Y directions and rotation around Z direction. $V_A$, $V_B$, $V_C$, and $V_D$ represent the velocity of four wheels Motor A, Motor B, Motor C, and Motor D, respectively. $V_x$ is the velocity of UGV along the x-axis, $V_y$ is the velocity of UGV along the y-axis, and $\omega$ is the angular velocity around the z-axis. Here, a is half the width of the UGV and b is half the length of the UGV.

When UGV goes along the x-axis, the following equation is obtained

$$
\begin{align*}
V_A &= -V_x \\
V_B &= +V_x \\
V_C &= -V_x \\
V_D &= +V_x
\end{align*}
$$

Figure 6. UAV flight mode control flow chart. UAV: unmanned aerial vehicle.

Figure 7. UGV hardware structure; UGV: unmanned ground vehicle.

OpenMV camera can acquire images in real time and transmit the processed coordinates to UGV, and then UGV tracks and moves to the target autonomously.
When UGV goes along the y-axis, the following equation is obtained
\[ VA = + Vy \]
\[ VB = + Vy \]
\[ VC = + Vy \]
\[ VD = + Vy \] (3)

When the car rotates around its geometric center, the following equation is obtained
\[ VA = - V_x + V_y - \omega(a + b) \]
\[ VB = + V_x + V_y - \omega(a + b) \]
\[ VC = - V_x + V_y + \omega(a + b) \]
\[ VD = + V_x + V_y + \omega(a + b) \] (5)

Using C language (the input parameters are \( V_x, V_y \), and \( \omega \)), the speeds of four motors are calculated and sent to the PID controller of UGV.

### Control mode of UGV

UGV has two control schemes: remote and automatic control. Each control scheme has two motion modes: speed and displacement modes. RC mode receives data from RC or mobile phone. Automatic mode receives the commands transmitted by other controllers through the serial port.

#### Speed control mode

When UGV is in speed control mode, the format of speed control data in each frame is shown in Table 1. According to Figure 8, \( Tx[1] \) controls the speed of the A motor; \( Tx[2] \) controls the speed of the B motor; \( Tx[3] \) controls the speed of the C motor; and \( Tx[4] \) controls the speed of the D motor.

#### Displacement control mode

When UGV is in displacement control mode, displacement is input. Mecanum UGV automatically performs kinematic analysis and converts displacement to speed. A 16-bit unsigned number synthesized by \( Tx[1] \) and \( Tx[2] \) controls the x-axis displacement; a 16-bit unsigned number synthesized by \( Tx[3] \) and \( Tx[4] \) controls the y-axis displacement; a 16-bit unsigned number synthesized by \( Tx[5] \) and \( Tx[6] \) controls the z-axis displacement. The format of coordinate-displacement control data in each frame is shown in Table 3.

---

**Table 1. Format of speed control data.**

| Data field | \( Tx[0] \) | \( Tx[1] \) | \( Tx[2] \) | \( Tx[3] \) |
|------------|-------------|-------------|-------------|-------------|
| Contents   | Mode        | Motor A speed control | Motor B speed control | Motor C speed control |
| \( Tx[4] \) | \( Tx[5] \) | \( Tx[6] \) | \( Tx[7] \) |
| Contents   | Motor D speed control | Default | Default | Direction control bit |

**Table 2. Format of \( Tx[7] \) in speed control mode.**

| Value | Contents                  | Value | Contents                  |
|-------|---------------------------|-------|---------------------------|
| 0     | Default                   | 0     | Default                   |
| 0/1   | Default                   | 0/1   | Default                   |

**Figure 8. Mecanum UGV motion pattern.**

UGV: unmanned ground vehicle.
Table 3. Format of displacement control data.

| Data field | Contents          | Tx[0] | Tx[1] | Tx[2] | Tx[3] |
|------------|-------------------|-------|-------|-------|-------|
| Contents   | Mode              | 0     | 0     | 0     | 0     |
|            | High 8 bits of x-axis displacement control | 0/1   | 0/1   | 0/1   | 0/1   |
|            | Low 8 bits of y-axis displacement control | 0     | 0     | 0     | 0     |
|            | High 8 bits of z-axis displacement control | 0     | 0     | 0     | 0     |
|            | Direction control bit | 0     | 0     | 0     | 0     |

Table 4. Format of Tx[7] in position control mode.

| Value | Contents          | 0 | 0 | 0 | 0 |
|-------|-------------------|---|---|---|---|
| Contents | Default | 0 | 0 | 0 | 0 |
| Value | 0 | 0 | 0 | 0 |
| Contents | Default | 0 | 0 | 0 | 0 |

Tx[7] is the direction control bit and has 8-bit data. Higher 5 bits are default and lower 3 bits control the direction of three axes (Table 4).

**UGV target recognition and tracking**

UGV also has an OpenMV camera to detect image information in real time and determines whether there is a Tag. If there is a Tag, the deviation between the center of the tag and the center of the image is calculated. Then, the deviation is taken as the input of incremental PID to control the rotation of the two-axis pan/tilt. At the same time, UGV plans the motion trajectory according to the current angle and distance from target. If no target is detected or target is lost, UGV stops moving and the two-axis pan/tilt performs a global search in 180° + 60°.

**Tracking scheme based on AprilTag**

AprilTag\textsuperscript{28} is an improved visual positioning system based on ARToolkit\textsuperscript{29} and ARTag.\textsuperscript{30} It is a visual reference library\textsuperscript{31} and widely used in robots and UAV positioning guide.\textsuperscript{32} AprilTag uses a simple Quick Response code (QR code) which has only 4- to 12-bit data and can be detected more robustly and from longer ranges.

AprilTag not only can identify and track the target but also can get the three-dimensional (3D) pose of the target. As long as the camera resolution, focal length, and the size of tag are known, the algorithm can identify the type, ID, distance, and attitude of the tag.

**Detection and identification of Tags**

The Tag is a quadrangle that is inner black and outer white, as shown in Figure 9.

The tag detection algorithm begins by computing the gradient at every pixel, including their magnitudes (Figure 10(a)) and direction (Figure 10(b)). Using a graph-based method, pixels with similar gradient directions and magnitude are clustered into components (Figure 10(c)). Weighted least square is used to fit these pixels of each component with a line segment (Figure 10(d)). The direction of the line segment is determined by the gradient direction so that segments are dark on the left and light on the right.

At this point, the Tag is transformed into a set of directed segments, and then the sequence of segments of the quadrilateral is calculated. The method used is based on a recursive depth-first search with a depth of 4.\textsuperscript{18}

**Calculation of the distance and angle from Tag to camera**

In homography transformation (one plane is mapped to another) and external parameter estimation, a 3 × 3 homography matrix (a conversion matrix when mapping, and the matrix is usually represented by \( H \)) needs to be calculated. It maps the coordinate system of the Tag to a 2D image coordinate system. Homography matrix is calculated by direct linear transform (DLT) algorithm. The position and orientation of the Tag require additional information, namely the focal length of camera and the physical size of Tag. The 3 × 3
homography matrix can be written as the product of a camera projection matrix $P$ with $3 \times 4$ order and an external parameter matrix $E$ with $4 \times 3$ order. The external parameter matrix is usually $4 \times 4$ steps, and each position satisfies $z = 0$ in coordinate system of Tag. Therefore, the coordinates of Tag can be rewritten as a 2D homogeneous point by deleting the third column of matrix $P$ to form a truncated external parameter matrix. The rotational component of $E$ is represented as $R_{ij}$ and the translational component is represented as $T_k$. $s$ is a scale factor

$$H = \begin{bmatrix} h_{00} & h_{01} & h_{02} \\ h_{10} & h_{11} & h_{12} \\ h_{20} & h_{21} & h_{22} \end{bmatrix} = sPE$$

$$= s\begin{bmatrix} f_x & 0 & 0 \\ 0 & f_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R_{00} & R_{01} & T_x \\ R_{10} & R_{11} & T_y \\ R_{20} & R_{21} & T_z \end{bmatrix}$$

Here, $h_{ij}$ is the element of homography matrix $H$. $f_x$ and $f_y$ are the focal lengths of camera, respectively. It is not possible to solve $E$ directly because $P$ is not full rank. By calculating the right side of equation (6), each $h_{ij}$ can be written into a set of equations

$$\begin{align*}
h_{00} &= sR_{00}f_x \\
h_{01} &= sR_{01}f_x \\
h_{02} &= sT_xf_x \\
&\ldots
\end{align*}$$

The elements of $R_{ij}$ and $T_k$ can be easily determined. Each column of the rotation matrix must be a unit vector, so the limitation can be satisfied by different $s$. Since the rotation matrix has only two columns, $s$ can be set as the geometric mean of the amplitude of matrix. Because the columns of rotation matrix must be orthogonal, the third column can be recovered by computing the cross product of two known columns.

The above DLT process and normalization process cannot guarantee that the rotation matrix is strictly orthogonal. So, to solve this problem, $R$ can be decomposed by polar coordinates to generate a Frobenius matrix norm with minimum error.

Finally, by homography matrix, the relative coordinate system of Tag is mapped to the coordinate system of image. Furthermore, the distance and angle from Tag to camera are finally obtained.

**Improved tracking based on color histogram**

Color histogram is a statistics of color distribution, and it is not affected by the changes of shape and attitude. Color histogram is used to improve the tracking method to get better stability and anti-occlusion ability. To reduce the influence of light changes, the color histogram is selected under the HSV (hue, saturation, value) color system.

The three components of HSV are quantified separately according to how they are sensitive to color changes. Suppose the values of the three components after quantization are \{0, 1, $L_H - 1\}$, \{1, $L_S - 1\}$, and \{1, $L_V - 1\}$, respectively, design a vector in the form of $[H, S, V]$, so its range is: \{0, 1, $L_H - 1\}$, \{1, $L_S - 1\}$, \{1, $L_V - 1\}$. Suppose the number of pixels of color $i$ is $m_i$, the total number of pixels of the image is

$$M = \sum_{i=0}^{ Lv-1} m_i$$

The probability $p_i$ of color $i$ is defined as a color histogram

$$p_i = \frac{m_i}{M}$$

Since the color histogram is a vector, the Bhattacharyya distance can be used as a measure of similarity of two histograms when tracking. The calculation of Bhattacharyya distance is

$$\rho(p, q) = \sum_{k=1}^{m} \sqrt{p(k) \cdot q(k)}$$

$$d(p, q) = \sqrt{1 - \rho(p, q)}$$

Here, $\rho$ is the Bhattacharyya coefficient of two histograms; $p$ is the histogram of target; $q$ is the histogram of template; and $d$ is the Bhattacharyya distance of two histograms. The smaller the value of $d$, the higher the similarity of the two histograms.

**Simulation and experiment**

**Simulation of tracking target**

KCF is a common tracking algorithm and it is used to make comparison with AprilTag. KCF abstracts the tracking problem into a linear regression model. To adapt to the deformation of target, KCF is modeled by ridge regression that has the feature of regularization. The objective function of the ridge regression is

$$\min_w \sum_{i} (f(x_i) - y_i)^2 + \lambda ||w||^2$$

Here, $\lambda$ is a regularization parameter. Calculate the derivative of equation (12) about $w$ and let it be equal to zero. So the extreme value is

$$w = (X^T X + \lambda I)^{-1} X^T y$$

In equation (13), each row of $X$ represents $x_i$, $y$ is a Gaussian regression label, and $I$ is an identity matrix. Equation (13) can be written as a complex domain form

$$w = (X^H X + \lambda I)^{-1} X^H y$$
where $X^H$ is a complex conjugate transpose matrix. To avoid calculating the inversion of matrix and accelerate the calculation, equation (14) is transformed to Fourier domain

$$\hat{w} = \frac{\hat{x} \Theta \hat{y}}{\hat{x}^T \Theta \hat{x} + \lambda}$$

where $\hat{x}$ is a fast Fourier transform of $x$, and $\hat{y}$ is a fast Fourier transform of $y$, and $\Theta$ represents the Hadamard product.

The simulation is carried out in a computer that has Intel Core i5-7300HQ CPU with 8GB memory. The operating system is 64bit Windows10 that has installed MATLAB 2016a and OpenMV IDE. For comparison, both algorithms select tag36h11_1 as the target.

1. Comparison between KCF and AprilTag

After running KCF, select the tag36h11_1 tag as the initial frame (Figure 11(a)), and AprilTag also selects the tag36h11_1 tag as the tracking target (Figure 11(b)).

After the selection of initial frame, both algorithms can accurately identify and track the target. To improve the running speed of AprilTag in embedded devices, the resolution is appropriately reduced and there is more noise (Figure 11(b)). However, it does not affect the accuracy of recognition.

Figure 12 shows the results of KCF and AprilTag when occlusion occurs. When occlusion just occurs, KCF can still determine the probable location of the target (Figure 12(a)), but AprilTag will lose the target (Figure 12(b)). It is because that KCF uses real-time online training to handle occlusion problems. Once the target is occluded, AprilTag will lose some features, hence the tracking fails.

After occlusion for a short period of time, KCF loses target (Figure 12(c)) and AprilTag still tracks target (Figure 12(d)). The comparison in Figure 15 shows that AprilTag can re-identify and track the target as soon as occlusion is removed.

When UGV moves, the two methods are also tested. Both algorithms are able to identify and track the target in low speed. When the speed of UGV is fast or suddenly changes, KCF loses its target (Figure 13(a)), while the AprilTag performs well (Figure 13(b)). KCF uses a local search method, so it will lose the target with high speed.

Since KCF uses a local search method, when the target motion speed is too fast, exceeding the search range will cause the target tracking to fail.

2. Effect of color features on the AprilTag

As can be seen from Figure 12(b), once the Tag is occluded, the target will be lost. This can seriously
affect the tracking of moving targets. Color information around the tag is added, to prevent the loss of target caused by occlusion. Figure 14 shows the comparison of tracking with or without the color feature.

When the Tag is occluded, AprilTag improved by color feature will identify the blue information around the Tag to ensure that UGV is always tracked until the Tag appears again. When the Tag is visible, it is automatically switched to the original method. Since color feature is easily affected by environmental factor, it is only used as a supplementary to tracking.

Figure 12. Occlusion in (a) KCF and (b) AprilTag and (c) KCF loses target after occlusion and (d) AprilTag still tracks target after occlusion. KCF: kernel correlation filter.

Figure 13. Moving target tracking of KCF and AprilTag: (a) target with high speed in KCF and (b) target with high speed in AprilTag. KCF: kernel correlation filter.

Tracking experiment

Tracking experiment of UGV. Since UGV uses flat view to track target and target moves in a complex background, the algorithm is original AprilTag without adding color feature. If target is lost, UGV will stop moving. Gimbal camera will rotate to making global search until camera detects the target again.

Figure 15 shows that UGV can adjust the speed according to the distance from target, so the distance between UGV and target is basically unchanged.
Figure 15(b)–(d) is a set of turning test. Because UGV is equipped with four omnidirectional wheels and the rotation angle of pan/tilt is $\frac{180^\circ}{\pi}$, tracking experiment of UGV is satisfactory.

Collaborative tracking experiment of UAV/UGV. Since UAV uses top view and the color features around UGV are not too complicated, the algorithm adopts AprilTag improved by color feature. When the target is lost, UAV will enter hover mode and wait for the target to reappear or the RC command.

In initial state, UAV parks on UGV as shown in Figure 16(a). In Figure 16(b), UAV takes off automatically and flies to a predetermined altitude. Figure 16(c) shows that the proposed method can help UAV/UGV heterogeneous system to complete the moving target tracking task successfully in good lighting. Figure 16(d) shows that the tracking effect is still good at night, when the lighting conditions are poor, and even total reflection occurs.

**Conclusion**

The focus of this paper is on the integration of UAV/UGV system. A heterogeneous collaborative system consisting of UAV and UGV is designed. Then, an analog RC based on the SBUS protocol is proposed for UAV. Next, a UGV with omnidirectional wheel is also
designed for the heterogeneous system. To improve the effectiveness and accuracy of tracking, a tracking scheme based on AprilTag is studied. By simulation and experiment, the proposed heterogeneous collaborative system can realize the real-time tracking of moving target and the conclusions are as follow:

**Figure 16.** Tracking experiment of UAV/UGV heterogeneous system: (a) initial state, (b) automatic take off, (c) tracking target in good lighting, and (d) tracking target in bad lighting.

UAV/UGV: unmanned aerial vehicle/unmanned ground vehicle.

**Figure 17.** Tracking experiment of UAV/UGV heterogeneous system.

UAV/UGV: unmanned aerial vehicle/unmanned ground vehicle: (a) Before pedestrian is occluded (b) Pedestrian is occluded (c) After pedestrian is occluded (d) Continue to track the pedestrian.
1. Analog RC based on the SBUS protocol can maintain the original stability of UAV and reduce the workload of developing.
2. UGV has a variety of control modes to adapt to different tasks. Omnidirectional wheels enable UGV to meet multiple challenges from various environments.
3. The tracking scheme based on AprilTag runs well in embedded devices. The effectiveness and accuracy are satisfied. The scheme is suitable for UAV/UGV heterogeneous system to track moving target.

Further work
The main contribution of this paper is the establishment and verification of the system that can provide a physical experiment platform for different control and image processing methods. Next, we will experiment with several tracking methods in an unknown complex outdoor environment. In future, a method of separating image acquisition and image processing is being studied and it will indirectly realize the complex tracking algorithm in embedded device. Now, we have basically implemented the tracking based on arbitrary features, and Figure 17 shows some preliminary results which still needs some improvement.

In addition, the proposed heterogeneous system is being improved and further tested by a new structure that consists of a tethered UAV and a sport utility vehicle (SUV). In Figure 18, the tethered UAV has been developed and tested. The tethered UAV is a four-axis eight-paddle UAV which has stronger power and more stable flight. The electro-optical pod is equipped with an HD sport camera and tethered cable is responsible for electric energy transport and data return. In the future, the modified SUV will be added to the heterogeneous system for more actual experiments.

Acknowledgement
It is worth mentioning that the author used the method and works of this article to participate in the “2018 the 13th China Graduate Electronic Design Competition” and won the first prize in the National Finals. The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

Declaration of conflicting interests
The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding
The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by the National Natural Science Foundation of China under Grant 61973222, 61503255, and 61906125; Aeronautical Science Foundation of China under Grant 2016ZC54011; and Natural Science Foundation of Liaoning Province under Grant 2019-ZD-0247.

ORCID iD
Xiao Liang https://orcid.org/0000-0002-5162-1240

Supplemental material
Supplemental material for this article is available online.

References
1. Manjanna S, Hansen J, Li AQ, et al. Collaborative sampling using heterogeneous marine robots driven by visual cues. In: 14th conference on computer and robot vision, Edmonton, AB, Canada, 16–19 May 2017.
2. Hager CE, Kwon H, Zarzhitsky D, et al. A cooperative autonomous system for heterogeneous unmanned aerial and ground vehicles. In: Infotech at aerospace 2011, St. Louis, MI, 29–31 March 2011.
3. Grocholsky B, Keller J, Kumar V, et al. Cooperative air and ground surveillance: a scalable approach to the detection and localization of targets by a network of UAVs and UGVs. IEEE Robot Autom Mag 2006; 13(3): 16–25.
4. Wenzel KE, Masselli A and Zell A. Automatic take off, tracking and landing of a miniature UAV on a moving carrier vehicle. J Intell Robot Syst 2011; 61(1–4): 221–238.
5. Hui C, Yousheng C, Xiaokun L, et al. Autonomous takeoff, tracking and landing of a UAV on a moving UGV using onboard monocular vision. In: Proceedings of the 32nd Chinese control conference. Xi’an, China, 26–28 July 2013.
6. Sarras I, Marzat J, Bertrand S, et al. Collaborative multiple micro air vehicles’ localization and target tracking in GPS-denied environment from range–velocity measurements. Int J Micro Air Veh. Epub ahead of print 4 January 2018. DOI: 10.1177/1756829317745317.
7. Ramya P and Rajeswari R. A modified frame difference method using correlation coefficient for background subtraction. Procedia Comput Sci 2016; 93: 478–485.
8. Ghassabeh YA and Rudzicz F. The mean shift algorithm and its relation to kernel regression. Inform Sciences 2016; 348: 198–208.
9. Chen TD, Hu J, Lu C, et al. Moving target tracking using sparse optical flow method. Adv Mat Res 2013; 718–720: 2335–2339.
10. Zhang B. Moving target detection and tracking based on CamShift algorithm and Kalman filter in sport video. Int J Performability Eng 2019; 15(1): 288–297.
11. Dufek J and Murphy R. Visual pose estimation of rescue unmanned surface vehicle from unmanned aerial system. Front Robot AI 2019; 6: 42.
12. Bradski GR. Computer vision face tracking for use in a perceptual user interface. Intel Tech J 1998; 2: 214–219.
13. Xu LB, Tang S, Yang J, et al. Robust autonomous car-like robot tracking based on tracking-learning-detection. Appl Mech Mater 2014; 687–691: 564–571.
14. Zhang L and Suganthan PN. Robust visual tracking via co-trained kernelized correlation filters. Pattern Recogn 2017; 69: 82–93.
15. Danelljan M, Bhat G, Khan FS, et al. ECO: efficient convolutional filters for tracking. In: IEEE conference on computer vision and pattern recognition, Honolulu, HI, 21–26 July 2017.
16. Danelljan M, Robinson A, Khan FS, et al. Beyond correlation filters: learning continuous convolution operators for visual tracking. In: European conference on computer vision, 11–14 October 2016, https://arxiv.org/pdf/1608.03773.pdf
17. Qu Z, Lv X, Liu J, et al. Long-term reliable visual tracking with UAVs. In: IEEE International Conference on Systems, Man, and Cybernetics, Banff, AB, Canada, 5–8 October 2017.
18. Olson E. AprilTag: a robust and flexible visual fiducial system. In: IEEE international conference on robotics and automation, Shanghai, China, 9–13 May 2011.
19. Wang J and Olson E. AprilTag 2: efficient and robust fiducial detection. In: IEEE/RSJ international conference on intelligent robots and systems (IROS), Daejeon, South Korea, 9–14 October 2016.
20. Xiao X, Dufek J and Murphy R. Visual servoing for tele-operation using a tethered UAV. In: IEEE international symposium on safety, security and rescue robotics (SSRR), Shanghai, China, 11–13 October 2017.
21. Wang J, Sadler C, Montoya CF, et al. Optimizing ground vehicle tracking using unmanned aerial vehicle and embedded AprilTag design. In: International conference on computational science and computational intelligence (CSCI), Las Vegas, NV, 15–17 December 2016.
22. Liang X, Chen G, Wang J, et al. An adaptive control system for variable mass quad-rotor UAV involved in rescue missions. Int J Simul Syst Sci Tech 2016; 17(29): 221–227.
23. Ali ZA and Li X. Modeling and controlling of quadrotor aerial vehicle equipped with a gripper. Meas Control 2019; 52: 577–587.
24. Tran HK and Nguyen TN. Flight motion controller design using genetic algorithm for a quadcopter. Meas Control 2018; 51(3-4): 59–64.
25. Gülhan MM and Erbatur K. Kinematic arrangement optimization of a quadruped robot with genetic algorithms. Meas Control 2018; 51(9–10): 406–416.
26. Kun LJ. Advanced PID control and MATLAB simulation. Beijing, China: Electronic Industry Press, 2004.
27. Ma D, Xia Y, Shen G, et al. Flatness-based adaptive sliding mode tracking control for a quadrotor with disturbances. J Frankl Inst 2018; 355(14): 6300–6322.
28. Sagitov A, Shabalina K, Sabirova L, et al. ARTag, AprilTag and CALTag fiducial marker systems: comparison in a presence of partial marker occlusion and rotation. In: 14th international conference on informatics in control, automation and robotics, Madrid, 26–28 July 2017.
29. Khan D, Ullah S and Rabbi I. Factors affecting the design and tracking of ARToolKit markers. Comp Stand Inter 2015; 41: 56–66.
30. Celozzi C, Paravati G, Sanna A, et al. A 6-DOF ARTag-based tracking system. IEEE T Consum Electr 2010; 56(1): 203–210.
31. Bergamasco F, Albarelli A and Torsello A. Pi-Tag: a fast image-space marker design based on projective invariants. Mach Vision Appl 2013; 24(6): 1295–1310.
32. Hartley R, Kamgar-Parsi B and Narber C. Using roads for autonomous air vehicle guidance. IEEE T Intell Transp 2018; 19(12): 3840–3849.