2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

On a videoing control system based on object detection and tracking

Yanhao Ren, Yi Wang, Qi Tang, Haijun Jiang and Wenlian Lu

Abstract— **In this paper, we propose a camera control system towards occasionally videoing preassigned objects. Based on the technique of real-time visual detection and tracking, using the Kalman filter and re-identification (ReID), we propose continuous composition of lens, based on the atomic rules of shots, and give the trajectory planning of the camera, to generate the PID controller to the pan-tilt. By both simulation and emulation by frame-wise cropping of video clips, we illustrate the efficiency of this method. Based on this model, we design and produce an AI automatic camera for lively photography and clip videoing.**

I. Introduction

Automatic cameras, sometimes installed on the robots, have been widely used in industry to take photos and videos in the unmanned circumstance or replacing humans [12,14,15,16]. The camera is equipped with pan-tilts, which enables to rotate in Euler angles: yaw, pitch, and roll, remoted controlled by servo motors via interaction to the manipulator [8,12,13,16].

With the contemporary development of AI, a new technology of unmanned photography has emerged that enables the automatic camera to work with less human participation, which is a combination of deep learning algorithms, computational aesthetics, and moving servo control [8,12,13,14,15,16]. Videoing is a common request in diverse fields, but unmanned AI videoing raises technical challenges in numerous aspects of algorithms, control, and hardware [14], because real-time semantic analyses of the scene, precise servo control of the pan-tilt for trajectory plan to maintain good compositions are difficult.

In this paper, we propose a videoing control model towards an automatic AI camera system, which, as shown in Fig. 1, is composed of three physical components: the camera, the pan-tilt with three Euler angles, and AI edge computing device (ECD). The basic routine of the system is to detect and analyze the objects in the screen, update the estimated tracking result, match the detection with tracking, determine the Euler angles of the camera at the next frame according to shot composition rule, generate the trajectory of the camera from the current frame to the next frame, control the camera by exerting forces or moments by the hardware of the camera.

Related works

The past few years have witnessed the rapid progress of computer vision with deep learning. Among them, classification and detection are the most mature. There are a lot of models proposed for one-stage object detection algorithms, including the SSD [17] and the YOLO family [18,19,20], trained by open datasets with high accuracy using GPUs. Visual tracking, including single object tracking (SOT) and multiple object tracking (MOT), is a common yet tough task in computer vision. Particle filter [22] and Kalman filter [6,11] are fine models in tracking objects, but they may fail when the object has some abnormal motions, particularly for the case of MOT task with difficult to treat occlusions [5], when in crowded places. In order to conquer the problem, the re-identification (ReID) method has been proposed [6,9], combining the Kalman filter trackers [6] and some other matching models like Siamese FC [3], Siamese RPN [4]. However, it is usually a trade-off between the accuracy and the limit of speed [9].

The trajectory planning of the camera is essential for automatic photographing: The cameras are equipped on pan-tilts or quadrotors and the camera moves together with these devices from one place to another by a given trajectory. While deciding the states (positions, angles, fields of view) of the camera, [2] proposed offline methods using simulated annealing algorithm, and the location technology on tracking. These methods can guarantee the camera to move properly with the target. The trajectories are generated by interpolation between two camera states [13, 21, 23]. The motion of the camera should conform to the physics law by forces or moments by the motor servo [13].

Through the continuous motion of camera states, a current videoing system can be developed. [8,12,13,14] developed such a system, by setting a rough route for the targets beforehand. Another problem for videoing is to keep the targets in a pleasing position on the screen. With the atomic rules of shot, different camera poses are used while taking different types of shots [7,10,12]. There is also a wide range of composition rules, and the most widely used is the rule of the thirds [12]: *The focal point of a shot is placed at the intersection of horizontal and vertical lines splitting the screen into thirds.*

Our Contributions

We propose a model of videoing control system towards an unmanned AI camera system, with a combination of the technology of object detection, visual tracking, and pan-tilt servo control. The contributions include:

Yanhao Ren is with School of Mathematical Sciences, Fudan University, Shanghai, China, Shanghai Center for Mathematical Sciences, Fudan University, Shanghai, China. (e-mail: 18110840015@fudan.edu.cn).

Yi Wang, Qi Tang, Haijun Jiang are with Fantasy Power (Shanghai) Culture Communication Co., Ltd., Shanghai, China.

Wenlian Lu is with School of Mathematical Sciences, Fudan University, Shanghai, China, Shanghai Center for Mathematical Sciences, Fudan University, Shanghai, China, Key Laboratory of Mathematics for Nonlinear Science, Fudan University, Ministry of Education, Shanghai, China, Shanghai Key Laboratory for Contemporary Applied Mathematics, Fudan University, Shanghai, China.

This work was supported by the Shanghai Municipal Science and Technology Major Project under Grant 2018SHZDZX01 and ZJLab.

978-1-7281-6211-9/20/$31.00 ©2020 IEEE

1. The proposed model is towards unmanned camera system for occasionally shots, other than designed shot [12,13], remote control camera, or human-camera interaction system [7,10];
2. We propose algorithms of real-time continuous composition proposal based on visual tracking, other than the composition of targets with predetermined places or routes [8,12,13,14];
3. We also introduce on-line continually trajectory planning to balance the atomic rules of shot [1], continuous and energy/pan-tilt consumption, other than the offline method [2].

II. Mathematical Model and Methods

A. Framework of videoing control system

We are to take videos of moving objects based on the technologies of object detection, visual tracking, composition, camera control, and trajectory planning. The framework diagram of the videoing control system is shown in Fig. 1, which is composed of the following parts:

Visual detector. The visual detector is on either the lens of the camera or an auxiliary webcam installed on the pan-tilt, equipped with a real-time object detecting program, using the Darknet of YOLOv3 [20] in this paper, running on a GPU-based EDC. The camera lens or webcam periodically shots a picture of the current scene and send it to the EDC, which detects and localizes the preassigned object to be videoed.

Visual tracker. At each frame of the video, the position of the object in the next frame is estimated using the Kalman Filter [11], with necessary transformation of coordinates. Besides, we also employ the ReID algorithm (the Hungarian algorithm) using a similarity deep network [5,6], to handle the cases where the Kalman filter does not work.

Real-time composition proposal. After obtaining the estimated position of the targets, the camera is rotated to track the objects, and also keep the targets in appropriate positions of the camera using the atomic rules of composition.

PID control of pan-tilt. There may be multiple errors in camera control, caused by detection errors, rotation errors, etc. We consider the output error of the control system as the distance between the targets' position and their expected position on the screen. The error is also used for the control of the camera using the PID control.

B. Coordinate transformation

In this system, the camera pan-tilt is fixed and we present the following hypotheses: i) The camera could only rotate but not move position by itself. In the world coordinate, we set the optimal center of the camera as the coordinate origin, and the target height minus the camera height is denoted as h ; ii) One side of the camera is always parallel to the ground.

Three coordinate spaces are defined as follows:

World coordinate. The x-y plane of the world coordinate is parallel to the ground and the z-axis is vertical to the ground.

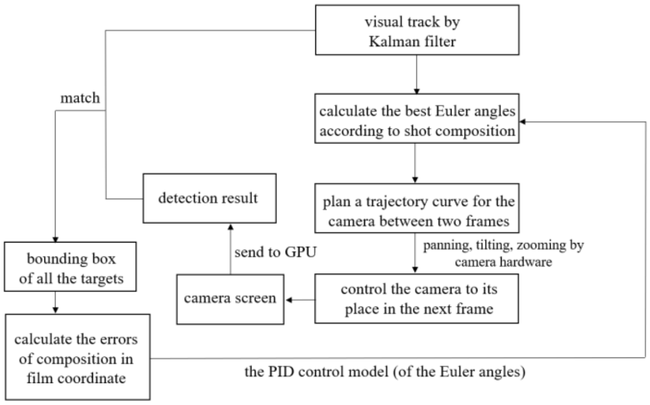
visual track by Kalman filter
match
calculate the best Euler angles according to shot composition
detection result
plan a trajectory curve for the camera between two frames
panning, tilting, zooming by camera hardware
send to GPU
bounding box of all the targets
camera screen
control the camera to its place in the next frame
calculate the errors of composition in film coordinate
the PID control model (of the Euler angles)

Figure 1. A framework diagram of the control system.

Camera rotation coordinate. The z-axis is the optic axis of the camera coordinate. The x-axis is parallel to the ground, and y-axis is generated by cross product. A relationship between camera coordinate and world coordinate is shown in Figure 2.

Film coordinate and pixel coordinate of the lens. Both the film and lens pixel coordinate are 2D with x-axis parallel to the ground and y-axis vertical to x-axis in the lens plane.

Denote (x_w, y_w, z_w) , (x_c, y_c, z_c) , (x_f, y_f)

as the world, camera, and film coordinates. The transformation between the world coordinate and camera coordinate is by formula (1) and (2).

$$(x_w, y_w, z_w) = (x_c, y_c, z_c)\mathbf{R}^T \tag{1}$$

where

$$\mathbf{R} = \begin{pmatrix} -\sin\varphi & -\cos\varphi\cos\theta & \cos\varphi\sin\theta \\ \cos\varphi & -\sin\varphi\cos\theta & \sin\varphi\sin\theta \\ 0 & \sin\theta & \cos\theta \end{pmatrix} \tag{2}$$

In Eq. (2), angles (θ, φ) are to be controlled, where θ is 90 degrees minus the pitch angle and φ is the yaw angle.

The transformation between the camera coordinate and film coordinate is by formula (3).

$$x_f = f\frac{x_c}{z_c}, \quad y_f = f\frac{y_c}{z_c} \tag{3}$$

where f is the focal length of the camera. The transformation between film coordinate and lens pixel coordinate is by translation and scaling.

According to (1) ~ (3), it can be seen that if one of x_w, y_w, z_w is known, the transformation among all these coordinate spaces has a unique solution.

C. Object visual tracking

Object tracking is conducted by real-time detection and the Kalman Filter. Herein, we take human bounding boxes with their confidence score of detection larger than a threshold. Kalman Filter for tracking boxes is a 7-dimensional vector, $\mathbf{X} = (x, y, s, r, \dot{x}, \dot{y}, \dot{s})$, where (x, y) is

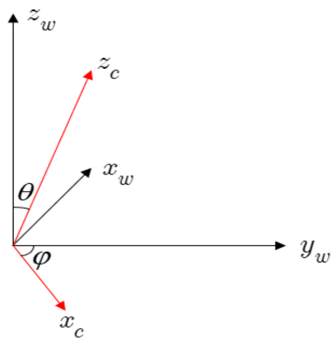

Figure 2. Relationship between the world and the camera rotation coordinates.

the coordinate of the center of the bounding box, s is the multiplication of the height and the width of the tracking box and r is the division between them.

We train a similarity deep network for ReID features. All matching is conducted by the Hungarian algorithm, and matching is considered successful only if the matching score is larger than a threshold. We utilize a similar pipeline as [6], shown in Algorithm 1, recursing from frame t to frame $t+1$.

Let $\{T_i^t\}$ be the tracking box set, $\{D_i^t\}$ the detection box set, $\{c_i^t\}$ the time period that the tracking box is not matched, $ID\text{track}(t)$ and $ID\det(t)$ the sets of tracking and detection boxes IDs at frame t respectively, T the set of the estimated tracking boxes, D' the set of detection boxes before matching, D the set of the matched detection boxes, and $\{D''_m\}_{m=t-\tau+1}^{t}$ the detection boxes of ID m from frame $t-\tau+1$ to frame t. Here, the superscript stands for the frame number and the subscript of the detection/tracking box ID except for special statements. All the sets mentioned above can be dynamical, during the control process.

D. Composition

According to the atomic rules of shot, we make the composition of the photograph by utilizing the rule of the thirds. When single object videoing (SOV), Figure 3(a) illustrates the basic composition, where the crosses stand for the expected positions of the targets. While multiple object videoing (MOV), the average position of the objects should be in the middle of the screen in the vertical side and should obey the rule of the thirds in the horizontal side, as shown in Figure 3(b), where the cross stands for the expected average position. Moreover, for MOV, we expect the target on the left side and the right side lie in the place in the green lines, whose distance to the edge of the screen is 1/6 of the length of the horizontal side of the camera screen.

While dealing with bounding boxes, the vertical side of the composition is simple by keeping the vertical bisector of the bounding box coincides with the vertical lines of the screen. In the horizontal side, we expect the center of the person's head to be at the crosses, but due to the reason that

Algorithm 1. The object tracking pipeline at frame $t+1$

```
Input: ${T_i^t}_{i \text{ in } ID\text{track(t)}}$, $\bigcup_{m=t-\tau+1}^{t}\{D_i^m\}_{i \text{ in } ID\det\text{(t)}}$, $\{c_i^t\}$ .
Output: $T=\{T_i^{t+1}\}_{i \text{ in } ID\text{track(t+1)}}$, $D=\{D_i^{t+1}\}_{i \text{ in } ID\det\text{(t+1)}}$, $\{c_i^{t+1}\}$ .
$D, D', \{D''_m\}_{m=t-\tau+1}^{t}, T \leftarrow \phi$, $E \leftarrow \{i$ for $i$ in $ID\text{track(t)}\}$
Run the Kalman Filter, estimate the tracking boxes in frame $t+1$ ( $T \leftarrow \{T_i^{t+1}\}_{i \text{ in } ID\text{track}(t)}$ )
Generate all detection boxes in frame $t+1$ without ID ( $D' \leftarrow \{D_j'^{t+1}\}_{j=1}^{N}$ )
                                                      /*here $j$ is not the ID*/
Use $IOU$ score to match $T$ with $D'$
for $j=1$ to $N$ do
    if $D_j'^{t+1}$ is matched with $T_n^{t+1}$ then
        /*give the detection box an ID and add it to $D$ */
        $D_n^{t+1} \leftarrow D_j'^{t+1}$, $D \leftarrow D \cup \{D_n^{t+1}\}$,
        $D' \leftarrow D' \setminus \{D_j'^{t+1}\}$, $E \leftarrow E \setminus \{n\}$
    end if
end for
/*take out the unmatched IDs and their detection boxes in previous $\tau$ frames*/
for $m=t$ downto $t-\tau+1$ do
    for $k$ in $E$ do
        $D''_m \leftarrow D''_m \cup \{D_k^m\}$ (if $D_k^m$ exists)
    end for
end for
/*Use the ReID feature for matching*/
for $m=t$ downto $t-\tau+1$ do
    /*if all the detection boxes are matched, end the loop*/
    if $\#D'=0$ then
        break
    end if
    Use ReID feature to match $D'$ with $D''_m$
    for the detection boxes in $D'$ do
        if $D_j'^{t+1}$ is matched with $D_n''^{m}$ then
            $D_n^{t+1} \leftarrow D_j'^{t+1}$, $D \leftarrow D \cup \{D_n^{t+1}\}$,
            $D' \leftarrow D' \setminus \{D_j'^{t+1}\}$, $E \leftarrow E \setminus \{n\}$
        end if
    end for
end for
Give all the remaining boxes in $D'$ a new ID and add them to $D$ and $T$
if $\#D'>0$ then
/*stop the Kalman Filter if it is not matched for a continuous time of $\tau$ */
for $k$ in $E$ do
    $c_k^{t+1} \leftarrow c_k^t + 1$
    if $c_k^{t+1} > \tau$ then
        $T \leftarrow T \setminus \{T_k^{t+1}\}$ /*stop the Kalman Filter*/
    end if
end for
for $k$ in $(ID\text{track}(t) \setminus E) \cup \{\text{new IDs}\}$ do
    $c_k^{t+1} \leftarrow 0$
end for
```

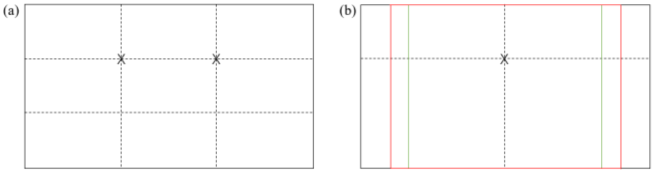

Figure 3. The composition according to the atomic rules of shot. The crosses are the positions the target should be on the camera screen according to the rule of the thirds. (a) The rule of SOV task. (b) The rule of MOV task. The green lines are the expected position of the targets on two sides. When at least one of the targets is outside the red rectangle, the camera needs to rotate and change the field of view. The area of the green rectangle is 2/3 of the screen and the red one is 4/5 of the screen.

part of the human may not be in the screen (e.g. the camera only catches the upper part of the object's body), we utilize the following experience: i) In a full-length photo, the average ratio between the width and the height of the human bounding box takes around 3:10. ii) The ratio between the head of a human to his whole body is around 1:7. According to i) and ii), we can calculate how much proportion of the height of the bounding box should be above the horizontal lines.

E. Trajectory planning

Trajectory planning includes the rotation of angles and changing the field of view. We use a quadratic curve for the trajectory of angles, which means the acceleration is a constant, and a linear interpolation for the field of view. Trajectory planning is conducted using different methods in SOV and MOV tasks.

SOV trajectory planning model. We only use the estimate of the Kalman Filter in the next frame as the standard. The angles (θ,φ) are determined by the location of the bounding box and the field of view α is changed by a factor of s_{t+1}/s_t .We conclude the formula of Kalman Filter here in (4) and (5).

$$\begin{aligned} \hat{\mathbf{X}}_{t+1}^{-} &= \mathbf{A}\hat{\mathbf{X}}_t + \mathbf{w}_t \\ \mathbf{z}_{t+1} &= \mathbf{H}\mathbf{X}_{t+1} + \mathbf{v}_{t+1} \end{aligned} \tag{4}$$

where $\mathbf{X}_t = (x_t, y_t, s_t, r_t, \dot{x}_t, \dot{y}_t, \dot{s}_t)$ is the real state, $\hat{\mathbf{X}}_t$ is the posteriori state, and $\hat{\mathbf{X}}_t^{-}$ is the priori state. $\mathbf{z}_t$ is the measurement, $\mathbf{w}_t$ and $\mathbf{v}_t$ are noises with normal distribution, and matrices $\mathbf{A}$ and $\mathbf{H}$ are defined as (5), where Δt is the interval between two frames.

$$\mathbf{A} = \begin{pmatrix} 1 & 0 & 0 & 0 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & \Delta t & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & \Delta t \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad \mathbf{H} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \tag{5}$$

Moreover, in order to avoid too frequent and too rapid motion of the camera, we set thresholds for the velocity of parameters (angles and field view) to conform to the following rules: i) If the velocity in the trajectory of a parameter is always less than a threshold between two frames, this parameter will keep stationary. ii) If the part the velocity in the trajectory is larger than a threshold, we keep the velocity of that part at the threshold with no acceleration. Although using i) and ii) may cause error in composition, this method can help the video look pleasant, especially when the target is activating in situ.

MOV trajectory planning model. While tracking, the objects may move outside the screen or too close to the edge of the screen, so we turn to determine the parameters by solving the following optimization problem in the film coordinate:

$$\min_{\theta,\varphi,\alpha} \lambda_1 d^2(\bar{p}_f, p_f^E) + \lambda_2 (d^2(x_{Lf}, x_{Lf}^E) + d^2(x_{Rf}, x_{Rf}^E)) \tag{6}$$

where x_{Lf}, x_{Rf} are the film coordinate of the left side target and the right side target and x_{Lf}^E, x_{Rf}^E are their expected positions, which are the green lines in Figure 3(b). $\bar{p}_f$ is the average position of all targets and p_f^E is the expected position, which is the cross in Figure 3(b). λ_1 and λ_2 are two hyperparameters and $d(\cdot,\cdot)$ means the Euclidean distance.

In order to avoid too much motion of the camera, we set two red lines, which is also shown in Figure 3(b), and the distance of the red lines to the edge of the camera is 1/10 of the length of the horizontal side of the camera screen. If none of the targets is outside the red rectangle, the camera will keep stationary. If at least one of the targets is outside the red rectangle, we solve (6) at frame t and use the solution (θ,φ,α) at frame $t+1$. At any time when the camera needs to rotate, a trajectory is planned (also with thresholds for the velocity of parameters) for the camera.

F. Pan-tilt control

The control of the pan-tilt is a PID control system. The system is equipped in the servo motor beforehand. In this task, we only use the PI control, and the update of the angles is conducted using (7).

$$x_{PID}(t) = x(t) + K_p e(t-1) + K_I \sum_{i=1}^{t-1} e(i) \tag{7}$$

where $x = \{\theta,\varphi\}$, $e(\cdot)$ means the error in the film coordinate. The determination of hyperparameters K_p and K_I is using the general principle of PID parameter adjustment.

III. SIMULATION EXPERIMENTS

For simulation, we consider objects as points moving in the world coordinate. The height of each object is constant.

A. Simulation models

SOV model. We suppose that the single object is on a map with N preassigned destinations. We use the random waypoint model on the object, and especially, it does not wait at any destination and has a constant moving speed. During the motion, the camera tracks the object. We set $h = 0$, so that the angle θ is kept fixed. Also, the focal length of the camera is kept fixed.

MOV model. The camera is to track multiple objects, which are moving slowly and synchronously, for instance, multiple persons, who are in the same activity, such as dancing or playing badminton. In this model, we detect and track multiple moving objects, which start of their initial positions, and start a same normal brown motion, and an independent small noise is added to each of the objects, as in (8).

$$X_i(t) = X_i(0) + B(t) + \sigma_i(t) \tag{8}$$

where X_i is the position of the i -th object in the world coordinate, B is the uniform normal brown motion for all objects, and σ_i is the noise.

Simulation model of the pan-tilt control system. The motion of the pan-tilt control system is formulated by the Euler-Lagrange equation (9) and the variation of $\Theta = (\theta, \varphi)$ is conducted by control moments.

$$\mathbf{M}(\Theta)\ddot{\Theta} + \mathbf{V}(\dot{\Theta}, \Theta)\dot{\Theta} + \mathbf{G}(\Theta) = \mathbf{Q} \tag{9}$$

where $\mathbf{M}$ is the inertia matrix, $\mathbf{V}$ includes the centrifugal force and the Coriolis force, $\mathbf{G}$ is the gravity, and $\mathbf{Q}$ is generalized force. Moreover, the PID control system (7) is realized by the servo motor of the system.

B. Results of SOV simulation

We conduct the simulation experiments by MATLAB R2016a. We set $f = 0.1$, $N = 4$ destinations, which are chosen randomly in the range of [1,10] ×[1,10]. The size of the camera is set as 0.16×0.09. The time length of each experiment is 45 sec. The velocity of the object is 1/sec and the sampling interval Δt is 0.1 second. The added noise of the motion has a variance 0.01. The normal bias added to the real position is set as 0.01, which is considered as the detection result. The simulation is conducted independently for 50 times, both with and without the PID control. We set the camera at the correct place in the first frame according to the composition rule, and the expected position in the film coordinate is always (0.16/6, 0.09/6) during tracking. An example of the trace of the object and the Kalman Filter output is shown in Figure 4(a). Figure 4(b) shows the Kalman Filter error in meters with frequency. It is seen that the Kalman Filter works well for the prediction of the motion. Under the obedience of the rule of the thirds, the angle $\theta \approx 98.53^{\text{o}}$. In the PID control of the camera pan-tilt, the best hyperparameters are obtained: $K_p = 10.0$ and $K_I = 0.9$. Figure 4(c) and Figure 4(d) shows the trace in the film coordinate with and without PID control of the example. Note that there is no error in the y coordinate because θ is a constant. Figure 4(f) shows the screen space error in centimeters with PID control with frequency. It is clear that the PID control can keep the camera close to the right position, while much larger errors appear without PID control. The variation of the angle φ is shown in Figure 4(e).

C. Results of MOV simulation

In the MOV simulation task, we set three moving objects. The motion of the objects is simulated using formula (8). An example is shown in Figure 5(a). When the targets are outside the red lines of Figure 3(b), the optimization problem (6) is solved and the camera rotates and changes the field of view to the solution. In the example, the height of the three targets are $h_1 = 0.1$, $h_2 = 0$, and $h_3 = -0.1$. The time length is 10 seconds. The sampling interval, noise, normal bias is the same as the SOV simulation. Because the influence of angle θ is small, we still keep it constant. The variation of the focal length of the camera f and the angle φ is shown in Figure 5(c) and Figure 5(d). We can see that these two variations change only in a certain time, and most of the time they are kept constant. Figure 5(b) shows the position of the targets in the film coordinate. It is seen that all the targets during the time period can be caught without much motion of the camera.

IV. Emulation Experiments

In this section, we generate a new video clip by cropping an original clip frame by frame, via tracking certain preassigned persons in the video and conforming to the composition rule, based on the simulating camera pan-tilt.

A. SOV emulation

For SOV, the video we use is an episode from *Sherlock Season 4.* The results are shown in Figure 6 in series. The interval between two scenes is 0.5 second. In Figure 6, we only show the part that is in the scene of the videos. The video scene is in the left column and the camera screen is in the right column. We use the Darknet of YOLOv3 for detection.

In this task, it is clear that the field of view becomes larger as the bounding box becomes larger. Error of composition is caused by the error of the bounding box, and also the reason that the person is not always in the center of the bounding box in the vertical side.

B. MOV emulation

For MOV, we use a video of a Chinese-style dance. The interval between two scenes is 1 second. The results are shown in Figure 7 in series. Note that we only track the three persons in the middle. In the MOV task, when the targets are not so close to the edge of the camera screen, the camera remains stationary. We can see clearly that the camera changes the yaw angle in the fourth picture and changes the pitch angle in the fifth picture, and keep stationary in the remaining ones. The field of view is changed according to the composition, but not according to the sizes of the bounding box. Although the composition rule is not always conformed to, the new scene in the new video clip taken by the camera is acceptable.

V. An AI automatic camera system

Partially by this model, we have developed an AI camera system for occasionally photography and videoing. This

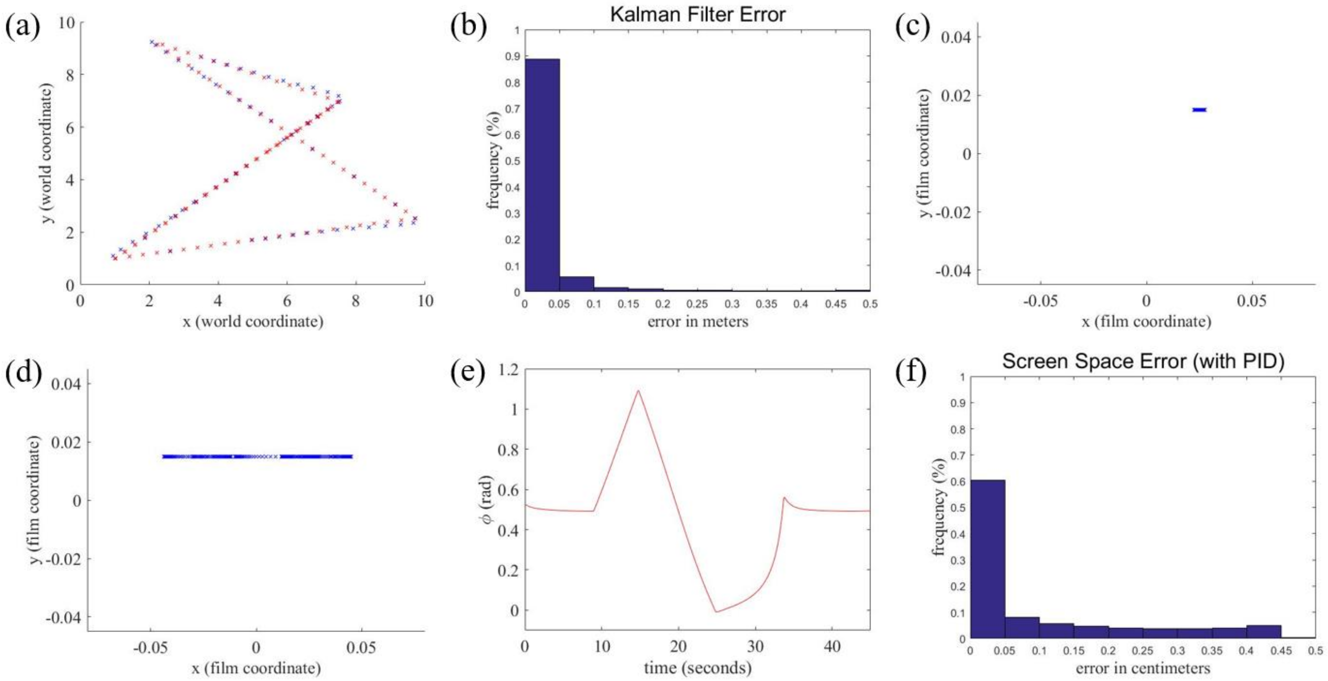

Figure 4. The result of the simulation of the SOV task. (a) An example of the motion model of the SOV task. The red crosses are real positions (in the world coordinate). The blue crosses are the estimated positions of the Kalman Filter. (b) The Kalman Filter error in meters after simulation for 50 times with frequency. The expected position on the camera screen is $(0.16/6, 0.09/6)$. The blue crosses in (c) and (d) show the position in the camera screen during the 45 seconds of the experiment. The results in (c), (d), and (e) are the result of tracking the motion in (a). (c) An example with PID control. (d) An example without PID control. (e) The variation of the angle φ . (f) The screen space error with PID after simulation for 50 times with frequency.

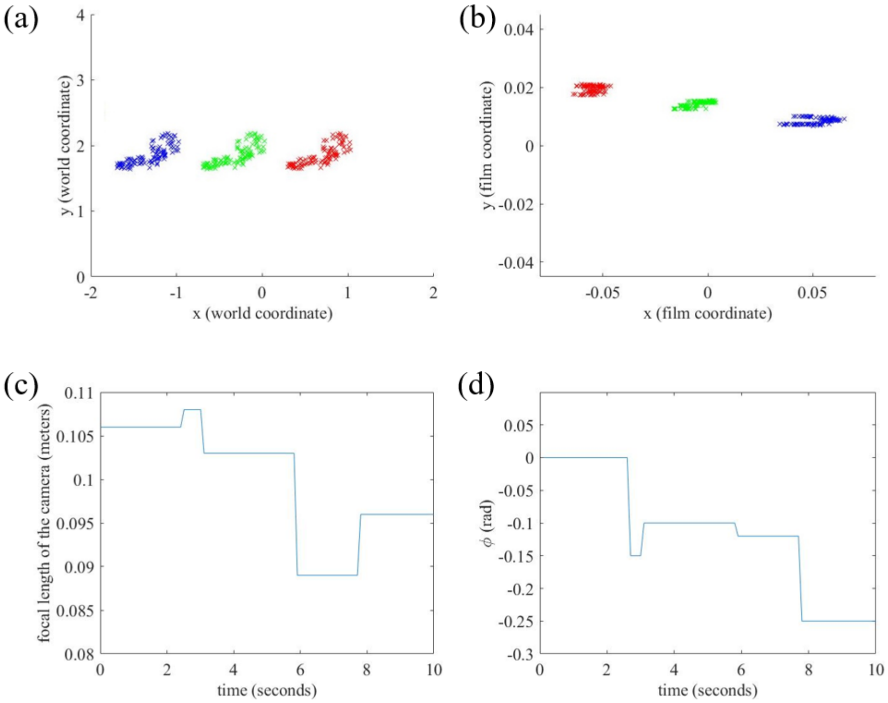

Figure 5. The result of the simulation of the MOV task. (a) An example of the motion model of the MOV task. The crosses of different colors are different tracking targets within the 10 seconds of the experiment. The targets are moving according to formula (8). (b) The positions of the targets in the film pixel coordinate. (c) The variation of the focal length of the camera. (d) The variation of the angle φ .

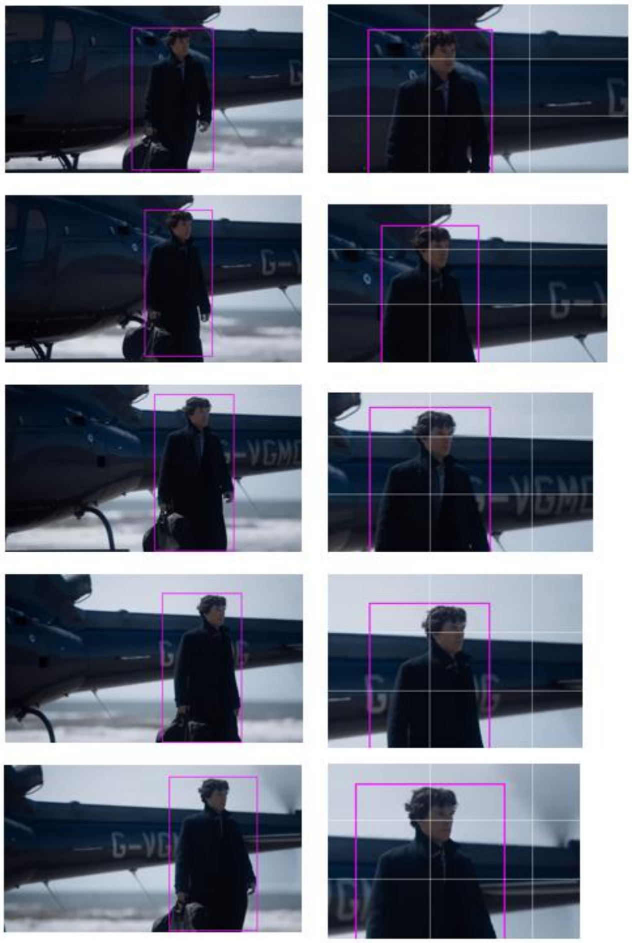

Figure 6. The result of SOV in tracking the hero of a movie episode. The interval of the image series is 0.5 seconds. The left column is the video scene and the detection results by YOLOv3. The right column is the camera screen. Part of the camera screen is not in the video scene, and we only show the part that is in the scene.

system includes this control method based on visual detection and tracking, camera device with pan-tilt, and AI edge computing device with readable storage medium. A flowchart of the system is shown in Figure 8. Real shots of our system are shown in the attachment video of this paper. The hardware structure of this system includes:

(i) **Optical lens.** The optical lens is of wide-angle to catch the image of the target, with the field angles 120°, 140°,170°, etc. We can layout three wide-angle lenses in different angles of the 360° circle, so that the detection can be conducted in a range of 360°.

(ii) **An image sensor.** The image sensor transforms the optimal signal to digital signals.

(iii) **An installation deck.** All components and modules are set on a special deck.

This system includes the following functions:

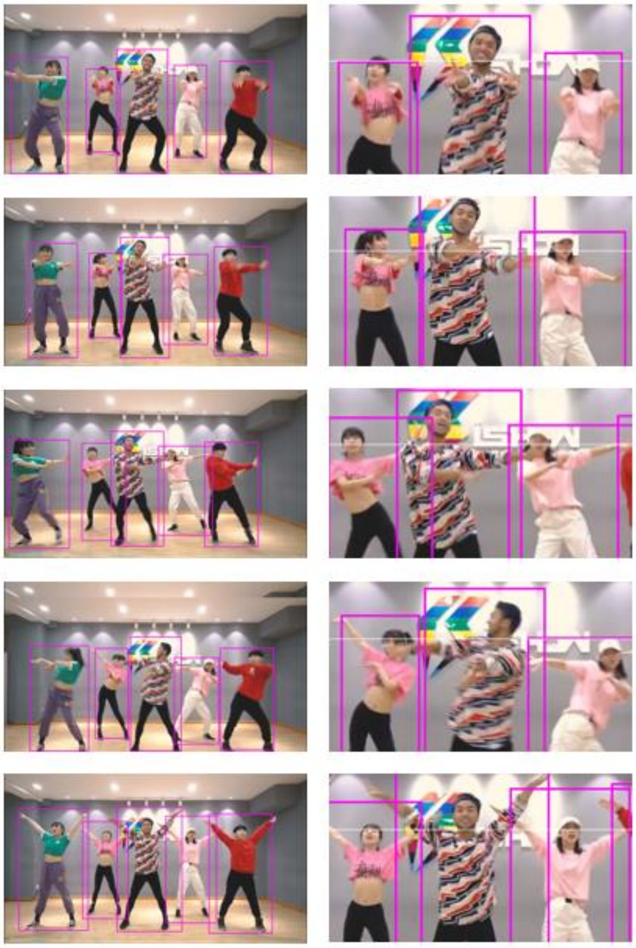

Figure 7. The result of MOV in tracking several dancers. The interval of the image series is 1 second. The left column is the video scene and the detection results by YOLOv3 and the right column is the camera screen. We only track the three persons in the middle.

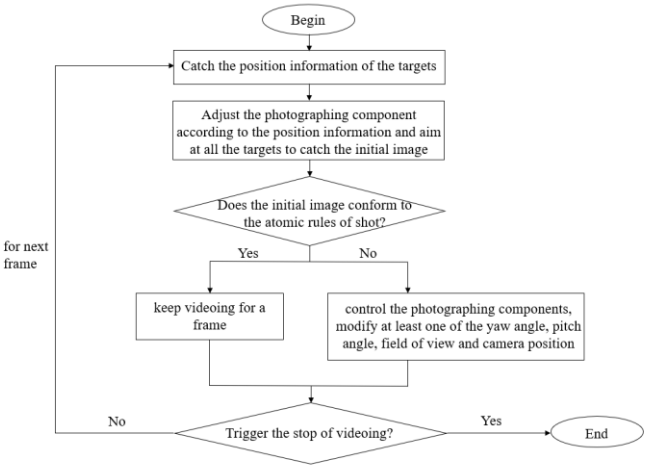
Begin
Catch the position information of the targets
Adjust the photographing component according to the position information and aim at all the targets to catch the initial image
Does the initial image conform to the atomic rules of shot?
Yes
No
keep videoing for a frame
control the photographing components, modify at least one of the yaw angle, pitch angle, field of view and camera position
for next frame
No
Trigger the stop of videoing?
Yes
End

Figure 8. A flow chart of our camera control system.

1) Catching the position information of the target.

2) Adjusting the photographing component according to the position information and aiming at the target.

3) Triggering the photographing component to catch the initial image of the target.

4) When the initial image conforms to the composition according to the atomic rules of shot, the photographing component can be triggered to execute the shooting of the target.

VI. Conclusions

We proposed a camera control system based on object detection and tracking. This control system includes rotation and zoom in-out of view of the lens, real-time continuous composition following certain atomic rules of shot, and PID control of the pan-tilt by servo motor. We conducted simulations and emulations on SOV and MOV problems with restriction of the motion of the camera to avoid vibration and overwhelming consumption motion. The PID control rule shows good performance in reducing the position error of the camera. Based on this model, we have also developed an automatic AI camera system. In this paper, we only take synchronously moving objects in the MOV task into considerations. The future orientation of our work may conduct more approaches of the camera control in MOV to improve the performance and take scenic sematic understanding into considerations as well.

References

[1] Arijon, D. “Grammar of the Film Language,” *New York: Communication Arts Books, Hastings House, Publishers.* 1976.

[2] Assa J, Cohen-Or D, Yeh I C, et al, “Motion overview of human actions,” *in ACM Transactions on Graphics, 2008, 27(5):*1.

[3] Bertinetto L., Valmadre J., Henriques, João F., et al. “Fully-Convolutional Siamese Networks for Object Tracking,” in *Computer Vision – ECCV 2016 Workshops. ECCV 2016.*

[4] Bo L., Yan J., Wu W., “High Performance Visual Tracking with Siamese Region Proposal Network,”, in *CVPR*, 2018.

[5] Bochinski E., Eiselein V. and Sikora T, “High-Speed tracking-by-detection without using image information,” in 2017 *IEEE* International Conference on Advanced Video and Signal Based Surveillance (*AVSS). IEEE*, 2017.

[6] Chen L., Ai H., Zhuang Z., and Shang C., “Real-Time Multiple People Tracking with Deeply Learned Candidate Selection and Person Re-Identification,” in *2018 IEEE International Conference on Multimedia and Expo (ICME) IEEE 2018*.

[7] Christianson D, “Declarative Camera Control for Automatic Cinematography,”. in *Proceedings os AAAI'96. 1996.*

[8] Dinh T., Yu Q. and Medioni G, "Real Time Tracking using an Active Pan-Tilt-Zoom Network Camera," in *IROS 2009. IEEE/RSJ International Conference on IEEE, 2009*.

[9] Feng W., Hu Z., Wu W., et al. “Multi-Object Tracking with Multiple Cues and Switcher-Aware Classification,” in *CVPR 2019.*

[10] He L.W., Cohen M.F., and Salesin D.H, “The Virtual Cinematographer: A Paradigm for Automatic Real-Time Camera Control and Directing,” in *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques, 1996.*

[11] Kalman, R. E. “A New Approach to Linear Filtering and Prediction Problems,” *Journal of Basic Engineering, 1960, 82(1):35.*

[12] Joubert N., L.E.J., Goldman D.B, Berthouzoz F., and Roberts M, “Towards a Drone Cinematographer: Guiding Quadrotor Cameras using Visual Composition Principles,” in *SIGGRAPH ASIA*, 2016.

[13] Joubert N., Roberts M., Troung A, et al, “An interactive tool for designing quadrotor camera shots.” in *SIGGRAGH ASIA, 2015.*

[14] Xie K, Yang H, Huang S, “Creating and Chaining Camera Moves for Quadrotor Videography.” in *SIGGRAPH 2018*.

[15] Levinson J., and Thrun S, "Automatic Online Calibration of Cameras and Lasers," in *Robotics: Science and Systems 2013.* 2013.

[16] Lino C, and Christie M, “Intuitive and efficient camera control with the toric space,” in *ACM Transactions on Graphics, 2015, 34(4):82:1-82:12.*

[17] Liu W., Anguelov D., Erhan D., et al. “SSD: Single Shot MultiBox Detector,” in *Computer Vision – ECCV 2016. ECCV 2016.*

[18] Redmon J., Divvala S., Girshick R., et al. “You Only Look Once: Unified, Real-Time Object Detection,” in *CVPR 2016.*

[19] Redmon J. and Farhadi A, “YOLO9000: Better, Faster, Stronger,” in *CVPR 2017.*

[20] Redmon J. and Farhadi A, “YOLOv3: An Incremental Improvement.” in *CVPR 2018.*

[21] Rhodes C, Morari M, Tsimring L S, et al. “Data-based control trajectory planning for nonlinear systems,” in *Physical Review E, 1997, 56(3):2398-2406.*

[22] Wang N. and Yeung D.Y, “Learning a Deep Compact Image Representation for Visual Tracking,” in *NIPS. Curran Associates Inc. 2013.*

[23] Yang C, Li Z, and Li J. “Trajectory Planning and Optimized Adaptive Control for a Class of Wheeled Inverted Pendulum Vehicle Models,” in *IEEE Transactions on Cybernetics, 2013, 43(1):24-36.*