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

A Target Tracking and Positioning Framework for Video Satellites Based on SLAM

Xuhui Zhao[1], Zhi Gao[1], Yongjun Zhang[1], Ben M. Chen[2]

Abstract— **With the booming development in aerospace technology, the video satellite which observes the live phenomena on the ground by video shooting has gradually emerged as a new Earth observation method. And remote sensing comes into a "dynamic" era with the demand for new processing techniques, especially the near-real-time tracking and geo-positioning algorithm for ground moving targets. However, many researchers merely extract pixel-level trajectories in post-processed video products, resulting in fairly limited applications. We regard the video satellite as a robot flying in space and adopt the SLAM framework for the positioning of ground moving targets. The designed framework is based on the representative ORB-SLAM and we make improvements mainly in feature extraction, satellite pose estimation, moving target tracking and positioning. We coordinate a moving fishing boat with GPS-RTK (Real-time Kinematic) devices and a video satellite observing it simultaneously for verification and evaluation of our method. Experiments demonstrate that our framework provides reasonable geolocation of the moving target in satellite videos. Finally, some open problems and potential research directions are discussed.**

I. INTRODUCTION

The rapid improvement in aerospace technology in recent years has brought some new methods for Earth observation, such as the nano video satellites, which have attracted widespread attention in remote sensing. Many commercial nano video satellites have been launched, such as the SkySat constellation launched by Planet Labs(first in 2013), the Jilin-1 constellation launched by Changguang Satellite Technology Co., Ltd.(first in 2015), and the Zhuhai-1 constellation launched by Obita aerospace technology Co., Ltd.(first in 2017). Video satellites usually gaze at a particular area on the ground for a period of time and output a video. Compared with static images and general videos captured by human beings or robots, it has some unique characteristics, such as high-definition (usually 4K resolution), high frame rate (usually more than 20 frames per second). And observed dynamic objects in satellite videos are usually with simple moving patterns (lines, smooth curves), small sizes (within a dozen pixels), and weak textures. All these signatures make the satellite video a brand-new object to tackle. Especially, the method for near-real-time tracking and positioning of moving targets plays a significant role in applications of satellite videos. Related research has emerged in recent years, but there is still a gap between present work and practical demands. The processing of satellite videos is generally as follows: satellites first capture the video of the ground and downlink it to the ground station for data production and analysis. Then some tracking algorithms are performed on video products for pixel-level trajectory of moving targets instead of the real geographic position. The whole procedure may take several hours to days, but few studies take into account the influence on applications introduced by this time delay. As aforementioned, it is more important to estimate the geolocation of moving targets in near-real-time rather than merely focusing on dynamic phenomena in satellite videos.

For near-real-time tracking and positioning of moving targets in video satellites, we regard the satellite in space as a flying robot and try to achieve our goal by combining the SLAM framework with satellite video processing methods. Considering the characteristics and limited computing resources of satellites, we choose simple and effective algorithms as possible. Modifications and improvements are made based on the representative ORB-SLAM [1]. Our work and contribution are mainly in the following three aspects:

- An efficient feature is modified for fast extraction and matching in satellite video frames.
- A practical transformation procedure is derived for estimating the absolute pose of video satellites.
- A novel tracking and positioning method is designed for moving targets in satellite videos.

The main components of this paper are as follows: Section 2 introduces the work related to this paper, and Section 3 is the specific description and algorithm of the framework. Section 4 is an experiment based on a satellite video, and detailed discussion of open issues and research directions is in Section 5. Finally, Section 6 is the summary and outlook.

II. RELATED WORK

Many tracking methods for moving targets in videos have been proposed in computer vision and intelligent perception. According to the sensor platform, it can be roughly divided into the drone platform and ground platform. On drone platforms, a framework for tracking and locating ground moving targets was proposed based on vision, AHRS (Attitude and Heading Reference System), and GNSS (Global Navigation Satellite System) [2]. However, it tracked and located ground moving vehicles with a small resolution of 480×320 pixels. Another tracking framework was designed based on the IMU and camera [3]. But it got the position of

[1]Xuhui Zhao, Zhi Gao and Yongjun Zhang are with School of Remote Sensing and Information Engineering, Wuhan University, 430079, Wuhan, China. zhaoxuhui@whu.edu.cn, gaozhinus@gmail.com, zhangyj@whu.edu.cn.

[2]Ben M. Chen is with Department of Mechanical and Automation Engineering, Chinese University of Hong Kong, 999077, China. bmchen@cuhk.edu.hk.

This work is partially supported by Peng Cheng Laboratory.

the moving target relative to a certain landmark rather than its real position. Compared with these methods, another tracking method was developed based on optical flow [4]. It did not require prior knowledge of moving targets but increased the calculation to a certain extent. Some other methods were also studied with different goals. For example, the location of the moving target relative to the drone was tracked for automatic navigation and obstacle avoidance (e.g., moving people), rather than focusing on the moving target and its real position [5]. And a tracking method focusing on the "interaction" between moving targets and drones was proposed similarly [6]. They made the drone constantly adjust its pose with the movement of the target to maintain the tracking state. As for ground platforms, a tracking method suitable for general videos based on SURF features was introduced [7]. And many methods for vehicle tracking on roads for autonomous driving were studied in [8], [9], [10]. Although video satellites and drones share similarities while they also have many differences, which may lead to a failure directly applying "drone methods""ggg". As mentioned earlier, moving targets in satellite videos usually have small sizes and simple moving patterns. Some tracking algorithms for satellite videos have been proposed considering these properties. It can be generally divided into four categories: background subtraction method, frame differencing method, optical flow method, and their variants. Based on foreground motion segmentation, a method for moving ship detection was proposed [11]. And a tracking method for vehicles was designed by fusing a kernel correlation filter and a three-frame difference method [12]. Based on optical flow and Gabor filters, a three-stage tracking method for the ship was also developed [13]. While a more accurate detection method for vehicles was proposed by combining the optical flow method, the HSV color space, and multi-frame difference methods at the same time [14]. Besides these methods, there are also some variant methods. For example, the improved tracking method considering the characteristics of moving targets in satellite videos and trajectory predictions [15], [16], the new DBM (Detection Box Model) method for moving target detection based on image enhancement and box detection [17]. However, related research was mostly performed on satellite video products, and only the pixel of the moving target was tracked instead of its geographic location. For example, the satellite video was used to extract vehicles, which shows its potential for traffic flow analysis [18]. But the work might be far from practical applications due to the time delay between processing and traffic.

Inspired by the tracking methods for drones, we combine the SLAM framework with satellite video processing methods. Specifically, we design a novel framework for near-real-time tracking and positioning of moving targets in satellite videos based on the ORB-SLAM and make improvements and additions in three aspects: feature extraction and matching, satellite pose estimation, moving target tracking and positioning. To our best knowledge, few studies combine these methods while some generally related work exists. For example, SLAM was applied for satellite proximity operations, docking process [19], [20], and landing on the moon combining with LiDAR [21]. But they are not designed for video satellites and positioning of moving targets.

III. The Proposed Framework

As shown in Fig. 1, the framework mainly includes three modules: pose estimation, target tracking, and geo-positioning. In pose estimation, we primarily apply visual SLAM methods to estimate the pose and motion of the satellite, providing necessary data for target positioning. In target tracking, an improved multi-template matching is performed for the pixel-level trajectory of the moving target. Finally, in geo-positioning, we calculate the three-dimension position of the target and transform it into the world coordinate based on estimated satellite poses and tracked pixels.

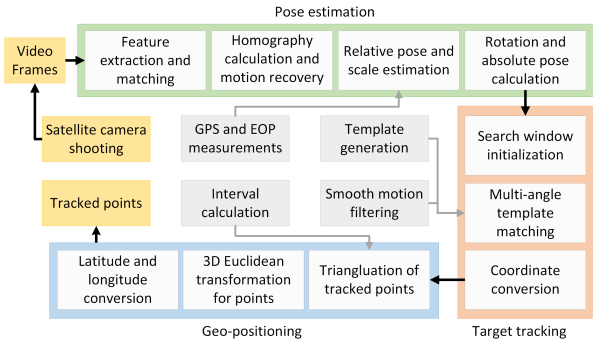

Fig. 1. The overview of the proposed ground moving target tracking and positioning framework for video satellites.

A. Feature Extraction and Matching

After a large number of experiments as shown in Table II, we find the SIFT [22] has a better performance and gets more matches with the same number of feature points compared with the ORB. And corners extracted by DoG (Difference of Gaussian) usually have a better distinguishability in satellite videos. While the ORB used in ORB-SLAM only calculates Oriented-FAST corners, resulting in higher efficiency. The slight change in illumination and geometry between video frames in a short period gives us the opportunity to find a trade-off between efficiency and performance for feature extraction and matching. We combine the advantages of SIFT and ORB, adopt DoG for extracting corners, and BRIEF (Binary Robust Independent Elementary Features) for description. The DoG kernel is defined in (1):

$$\begin{aligned} DoG &= G_{\sigma_1} - G_{\sigma_2} \\ &= \frac{1}{\sqrt{2\pi}}\left(\frac{1}{\sigma_1}e^{-(x^2+y^2)/2\sigma_1^2} - \frac{1}{\sigma_2}e^{-(x^2+y^2)/2\sigma_2^2}\right) \end{aligned} \tag{1}$$

Where σ_1 and σ_2 are standard deviations of the Gaussian distribution. For evenly distributed features, we discard the octree method used in ORB-SLAM due to a large amount of computation brought by the high resolution of satellite videos and divide the whole image into several blocks straightforwardly. And we use Hamming distance as the similarity metric for the BRIEF descriptor, which is the

same with ORB-SLAM. Finally, the Brute-Force matching is executed due to limited number of feature points.

B. Satellite Pose Estimation

Generally speaking, the pose of the satellite can be obtained from the carried attitude sensor and GPS sensor. However, these measurements may not be very accurate due to reasons such as extreme operational environment, accuracy limit in measurement, poor hardware quality in some low-cost micro/nano satellites. To solve this problem, we introduce the visual odometry in SLAM to video satellites for pose estimation. Sensors and the visual odometry work at the same time, and we fuse the measured data and the estimated pose together for a more accurate pose.

1) Relative pose estimation: Due to the small displacement of the video satellite between frames compared with scene depth(orbit altitude), which may lead to a little disparity and a decline in perception of subtle depth changes on the ground. We make a reasonable assumption according to common practice for some UAVs: we regard the local ground area captured by the video satellite as a plane, which means we ignore the variation of ground elevation. For example, video satellites generally run in an SSO (Sun-synchronous Orbit) around 500 km, the distance between frames is about 15.8 km with the first cosmic velocity if we estimate the pose every two second. The depth is around 31.6 times the displacement between two frames. The significance of this assumption is that we can simplify the process to a certain extent by directly using the homography H for frame-wise motion estimation instead of an essential matrix E or fundamental matrix F, avoiding the selection of F and H during monocular initialization in ORB-SLAM. On the other hand, we eliminate the scale ambiguity of the monocular SLAM by introducing the GPS measurements, as shown in Fig. 2. Usually, the GPS sensor measures the position and velocity of the satellite in WGS84 (World Geodetic System 1984), an ECEF (Earth-centered, Earth-fixed) system commonly used in cartography and satellite navigation. And there are different methods to recover the scale leverage on GPS measurements. One is that using a couple of measurements from GPS to get the scale factor and applying it for the rest of the time like the monocular initialization in ORB-SLAM. The other is using the measurements continuously for frame-wise pose estimation. While a visual loop closure for global optimization is not feasible to eliminate the sacle drift and error accumulation at present due to the unique one-way movement of satellites, we integrate the GPS measurements by the latter method.

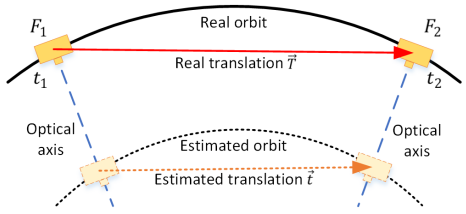

Fig. 2. The elimination of scale ambiguity by using GPS measurements.

The scale factor s between the estimated translation and real translation can be calculated by (2):

$$s = |\vec{T}|/|\vec{t}| \tag{2}$$

Where $\vec{T}$ is the real translation vector (red solid line in Fig. 2) and $\vec{t}$ is the estimated translation vector (orange dotted line in Fig. 2) of the satellite between frame F_1 and F_2. However, interpolation is needed for time synchronization between the camera and GPS sensor. For example, we obtain the GPS data every 80 frames in the Zhuhai-1 video satellite due to a frequency of 20 Hz in video shooting and 0.25 Hz in GPS measurement. For better results, we interpolate by Lagrange polynomials, as written in (3):

$$p_t = \sum_{j=0}^{k} p_j \left(\prod_{i=0, i\neq j}^{k} \frac{t - t_i}{t_j - t_i} \right) \tag{3}$$

Where p_t is the satellite position at a discrete time t.

2) Absolute pose transformation: For objects in space, the orbital motion is commonly described in J2000, an ECI (Earth-centered Inertial) system which does not move with Earth. After obtaining the relative pose between frames based on the homography H, we get corresponding pose in world coordinate by a series of transformations, involving the transformation from ECI system to ECEF system. Specifically, it takes four steps to transform the estimated relative pose between frames to the absolute pose in WGS84, as shown in Fig. 3. In this paper, R_A^B refers to the rotation from A to B or the pose of B with respect to A.

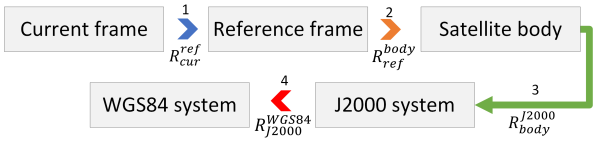

Fig. 3. The transformations for the absolute satellite pose estimation.

For step 1(blue), the rotation R_{cur}^{ref} is just the relative pose we estimated using the homography H. For step 2(orange), the rotation R_{ref}^{body} between the camera and satellite is determined by the satellite assembly and measured before the launch. For step 3(green), we can obtain the rotation R_{body}^{J2000} from the quaternion, which is measured in J2000 by attitude sensors on satellites. For step 4(red), the rotation R_{J2000}^{WGS84} between J2000 and WGS84 can be calculated by using the EOP (Earth Orientation Parameters), which is a collection of parameters that describe irregularities in the rotation of Earth. And it is measured, predicted, and published regularly by specialized agencies such as IAU (International Astronomical Union). Finally, we can write all steps together in (4):

$$R_{cur}^{WGS84} = R_{J2000}^{WGS84} R_{body}^{J2000} R_{ref}^{body} R_{cur}^{ref} \tag{4}$$

C. Moving Target Tracking and Positioning

Considering the characteristics of satellite videos and conventional algorithms, we design a tracking and positioning method for large moving targets(such as planes, ships) in satellite videos. The flow is outlined in Fig. 4.

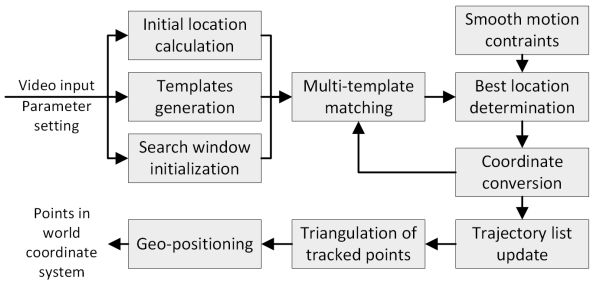

Fig. 4. The flow of our tracking and positioning method for moving targets in satellite videos.

We first employ the frame differencing and contour extraction to find the initial pixel location of the moving target and generate multiple templates from different angles. Based on this, the multi-template matching is performed in a small search window to find the best matching result. Then the local pixel result in the search window is converted to the global pixel result in the whole image. In this process, smooth motion constraints are also performed for checking. The tracking result is added to the trajectory if it meets the constraints. Finally, triangulation and geo-positioning are conducted based on the global pixel result and estimated motion of the video satellite.

1) Target tracking: We initially obtain the pixel location of the moving target based on the improved frame differencing method, and further generate multi-angle templates by the contour extraction. The flow of the algorithm is shown in Fig. 5 (Here we generate eight templates from different angles). More specifically, we get a potential moving target mask based on frame differencing and filter the noise by erosion operations. Then we regard the result with max response as our object of interest and generate templates, because large moving targets (such as planes, ships) are very rare in single satellite video. However, it may lead to unstable initialization if the video contains many tiny moving targets such as cars, but it is not the case we design the method for at present. A more detailed discussion is in Section 5.

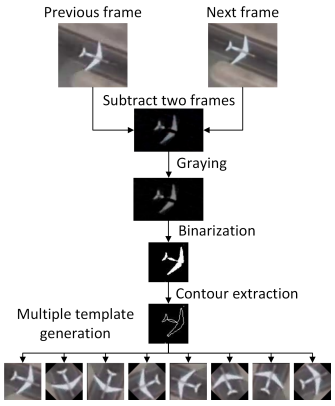

Fig. 5. The generation procedure for multi-angle templates.

For a more continuous contour, we set $0.8G_{max}$ as the empirical binarization threshold after a large number of experiments, where G_{max} is the maximum gray-scale value in a search window. And we adopt correlation coefficients as the similarity metric for matching. The evaluation function $R(x,y)$ between the template and image is defined in (5):

$$\begin{cases} T'(x,y) = T(x,y) - \frac{\sum_{x',y'} T(x',y')}{w \times h} \\ I'(x,y) = I(x,y) - \frac{\sum_{x',y'} I(x',y')}{w \times h} \\ R(x,y) = \sum_{x',y'} T'(x',y') \times I'(x+x',y+y') \end{cases} \tag{5}$$

Where $T(x,y)$ and $I(x,y)$ represent the template and image respectively. w and h are the width and height of the template. A larger R means a better matching result. And we choose the best result from all templates. For a better tracking result and less computation, we also adopt the dynamic search window strategy, which means the search window for template matching in the next frame is determined by the previous tracking result, as written in (6):

$$\begin{cases} X_{i+1} = X_i + x_{ti} - d \\ Y_{i+1} = Y_i + y_{ti} - d \end{cases} \tag{6}$$

Where (X_i, Y_i) and (X_{i+1}, Y_{i+1}) are the pixels of the upper left corner of the *i-th* and $(i+1)$*-th* search window in the whole image. (x_{ti}, y_{ti}) is the local pixel of the matched target in this window. d is the window size. We also derived the smooth motion constraints based on the continuity of the physical motion on the ground, as written in (7). It indicates the maximum pixel range that an object can move between frames. The tracking result may be uncommon and wrong if it flashes outside this range.

$$Range_{max} = ceil\left[\frac{5v}{18fg}\right] \tag{7}$$

Where f is the FPS (Frames Per Second) in a satellite video; g is the GSD (Ground Sample Distance), the unit is $m/pixel$; v is not the real velocity but the common speed of the moving object set by user for determining the window size, the unit is km/h, some objects and $Range_{max}$ are shown in Table I. $ceil[\cdot]$ means rounding up the result. In actual application, it is achieved by comparing the Euclidean distance between two trajectory points with $Range_{max}$.

TABLE I
MAX RANGES OF SOME COMMON OBJECTS WITH GSD 2M, FPS 24.

Objects	Velocity (km/h)	$Range_{max}$
Cars in express way	120	1(0.69)
High-speed trains	350	3(2.02)
Cruise ship	30	1(0.17)
Planes	800	5(4.62)

As aforementioned, the method is designed for tracking large single moving target in satellite videos based on multi-template matching. It may fail under some conditions, such as the moving target with rapid movement changes, small sizes, weak contours, similar objects.

2) Positioning model: The geographic position of the target is usually solved by the collinearity equation, which needs iterations based on DEM (Digital Elevation Model) and parameters that maybe not easy to obtain [23]. Therefore, few researchers calculate the location during the tracking of moving targets. However, we achieve this by using the triangulation and rigid transformation based on the estimated satellite poses and tracked pixels, avoiding the dependence on DEM and collinearity equation.

The positioning model mainly includes two stages: converting the pixel point to the camera coordinate and then transforming it into the world coordinate. As shown in Fig. 6, at time t_1, the camera is at C_1 and shoots the reference frame. After a period, it is at C_2 by a rotation R and translation t and shoots the current frame. P_w^1 and P_w^2 are geographic positions of the ground moving target at t_1 and t_2. P_t^1, P_t^2 are the tracked pixels. P_c^1, P_c^2 are corresponding points in the camera coordinate. P_p^1 is the pixel in the current frame transformed from P_t^1 using R and t.

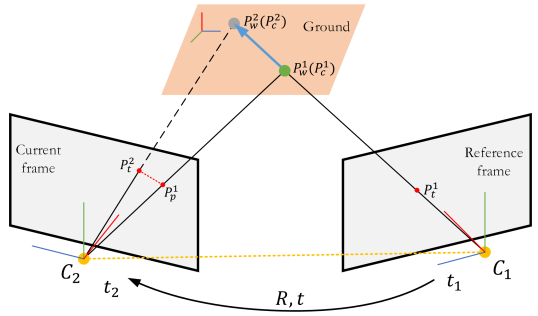

Fig. 6. The transformation of the positioning model for tracked points.

Based on the pinhole camera model, P_c^1 can be calculated theoretically by (8). Here P_t^1 is in homogeneous form.

$$P_c^1 = zK^{-1}P_t^1 \tag{8}$$

Where K is the intrinsic matrix of the camera, including the focal length and principal point. However, z is unknown and can not be solved by a single image. But with the estimated motion of the satellite (rotation R and translation t) and the tracked pixel P_t^1, P_c^1 can be calculated by triangulation. The only thing needs to be noticed is that the two tracked pixels P_t^1 and P_t^2 should not be used as the matching point pair for triangulation because of its inconsistent motion with the other static matching pairs. Here we obtain P_p^1 by transforming P_t^1 into the current frame according to the estimated R and t, and then use P_t^1 and P_p^1 as the corresponding points to calculate P_c^1. Besides, due to the high orbit and small parallax between video frames, we do not triangulate points frame by frame, but with a proper time interval to guarantee the accuracy of triangulation. We derive the relationship between the orbit altitude and corresponding intervals based on satellite orbit dynamics, as shown in (9):

$$\begin{cases} v_s = \sqrt{\frac{GM}{R+h}} \\ T_{int} = \frac{h}{rv_s} \end{cases} \tag{9}$$

Where the interval T_{int}(unit:*second*) is calculated by the orbit altitude h(unit:km), the ratio of max effective depth to baseline r and the velocity of the satellite v_s(unit:km/s). While v_s is calculated by the Newtonian gravitational constant G, the mass of Earth M, the radius of Earth. The remaining step is to transform the P_c^1 from camera coordinate into the world coordinate through rigid transformations. To facilitate continuous transformations, we choose the transformation matrix T to express rotation R and translation t together. For three-dimension transformations, T is a 4×4 matrix, and we use a similar notation with rotation for consistency, where T_A^B indicates the transformation from A to B. The transformation from the camera coordinate to WGS84 is shown in Fig. 7 (Rotation has been described previously and is omitted here).

Fig. 7. The transformation from the camera coordinate to WGS84.

GPS sensors measure the position of the satellite in WGS84, which is the translation t_{WGS84}^{body}. And T_{WGS84}^{body} can be obtained by combining t_{WGS84}^{body} and estimated R_{WGS84}^{body}. As mentioned before, both the translation t_{body}^{cam} and rotation R_{body}^{cam} between the camera and the satellite will be measured before the launch, which composes T_{body}^{cam}. The transformation matrix T_{WGS84}^{cam} can be written as (10):

$$T_{WGS84}^{cam} = T_{body}^{cam} T_{WGS84}^{body} \tag{10}$$

Finally, multiplying the P_c^1 in the camera coordinate and inverse transformation T_{cam}^{WGS84}, we can get the geographic position P_w^1 in the world coordinate, as written in (11):

$$P_w^1 = (T_{WGS84}^{cam})^{-1} P_c^1 = (T_{WGS84}^{body})^{-1} (T_{body}^{cam})^{-1} P_c^1 = T_{cam}^{WGS84} P_c^1 \tag{11}$$

After obtaining the P_w^1, it can be further transformed into the geodetic coordinate using (12):

$$\begin{cases} L = atan(\frac{Y}{X}) \\ B = atan(\frac{Z+e'^2 N sin^3\theta}{\sqrt{X^2+Y^2}-e^2 acos^3\theta}) \\ H = \frac{\sqrt{x^2+y^2}}{cosB} - N \end{cases} \tag{12}$$

Where $\theta = atan\frac{Z\cdot a}{\sqrt{X^2+Y^2}\cdot b}$, $N = \frac{a}{\sqrt{1-e^2 sin^2 B}}$, $e^2 = \frac{a^2-b^2}{a^2}$, $e'^2 = \frac{a^2-b^2}{b^2}$. (X,Y,Z) is the world coordinate of a certain point and (B,L,H) is the corresponding geodetic coordinate in latitude, longitude, and elevation. N is the prime vertical radius. e and e' are the first and second eccentricity of Earth ellipsoid. a and b are the long and short half axis of Earth ellipsoid. And we estimate the real heading and velocity based on positioning results while only pixel-level velocity can be obtained in tracking.

IV. Experiments

In this section, we first evaluate the feature extraction and matching performance of our method. Then we verify

the feasibility of our framework with a satellite video. Data specifications, error metrics, experiment set, and comparisons are detailed in related parts. Because it is not feasible to test our framework directly on a running video satellite, all experiments are performed on a consumer laptop configured with Intel Core I5-8300H CPU @ 2.3GHz, 8GB memory, and Ubuntu 16.04 operating system.

A. Evaluation of Feature Extraction and Matching

From the practical application, we evaluate our method in 200 frames of four different ground types(cities, harbors, etc.) and compare it with ORB and SIFT which are implemented in OpenCV Library, as shown in Table II. For fair comparison, we extract 10,000 keypoints for every frame without any acceleration tricks or hardware.

TABLE II

THE EVALUATION OF OUR METHOD IN 200 FRAMES WITH DIFFERENT GROUND TYPES (TIME UNIT: SECOND).

Feature	Mean time of feature extraction and description	Mean time of matching	Mean matching rate
ORB	0.21	0.43	21.79%
SIFT	2.00	5.13	56.34%
Ours	**1.97**	**0.41**	**51.86%**

Compared with ORB, we extract better features and achieve a higher matching rate. And we achieve similar performance with SIFT while much faster in matching due to the use of binary descriptors. In this scenario, our method is a good trade-off between efficiency and quality for feature extraction and matching in satellite videos.

B. Verification of Our Framework

1) Data specifications and metrics: To verify and evaluate our framework, we coordinated commercial satellite companies and personnel on the ground to observe the same moving target simultaneously. We rented a fishing boat and installed GPS-RTK (Real-time Kinematic) equipment on it sampling rate of 1 Hz for ground truth measurement and tracking. And Zhuhai-1 video satellite was used for shooting a video with a GSD of 1.98 m and FPS of 20, which runs at the orbit of 550 km at a speed of about 7.58 km/s. Yantai City was selected as the imaging area, which is located in the northeast of the Shandong Peninsula(China) with a range of 37.60N-37.56N in latitude, 121.37E-121.49E in longitude, as shown in Fig.8.

We assess the positioning accuracy by calculating the Root Mean Square (RMS) of absolute positioning error (APE) and relative positioning error (RPE) inspired by [24]. APE is the position difference of the estimated trajectory and ground truth, which reflects the absolute position accuracy in the long-term. And we subtract the mean positions from trajectories to remove the influence of absolute positioning and get the normalized estimated trajectory and ground truth with the mean position of $(0,0)$. RPE is calculated by the difference of them, which reflects the local consistency of positioning. APE and RPE are written in (13):

Fig. 8. One frame of the satellite video we use for experiments. (a) The fishing boat we used for tracking. (b) The multi-angle templates and corresponding matching results.

$$\begin{cases} APE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(p_i - g_i)^2} \\ RPE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}((p_i - \bar{p}) - (g_i - \bar{g}))^2} \end{cases} \tag{13}$$

Where p_i is the *i-th* estimated position and g_i is the *i-th* ground truth, $\bar{p}$ and $\bar{g}$ are the mean positions of the estimated trajectory and ground truth respectively. n is the number of observations. We also calculate the accuracy of estimated speed and orientation using (14):

$$Acc_{est} = 1 - \frac{Error_{est}}{Mean_{truth}} \tag{14}$$

Where $Error_{est}$ is the RMSE of the estimated speed or orientation, $Mean_{truth}$ is the mean value of the ground truth, and Acc_{est} is the accuracy of the estimation.

2) Experiment set-up and results: We extract 10,000 feature points in video frames for relative pose estimation and calculate the absolute pose using EOP published by IAU. Eight templates with a difference of 45 degrees between each other are applied for tracking. And we think the triangulation is reliable within the range of 40 times baseline, resulting in an interval of 1.81 s for tracking and positioning calculated by (9) and set to 2 s in practice. Finally, thirty frames are selected every forty frames while the whole experiment video contains about 1,500 frames(one minute). Absolute positioning results are shown in Fig. 9, where orange dots represent the estimated trajectory, and blue dots represent the ground truth. The absolute positioning error of the moving boat is summarized in Table III.

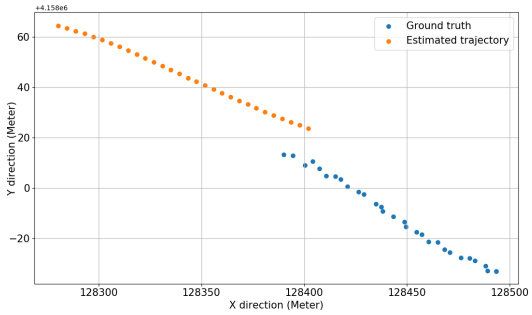

Fig. 9. Absolute tracking and positioning results of the fishing boat.

TABLE III

THE COMPARISON FOR APE & RPE OF THE MOVING FISHING BOAT USING OUR METHOD AND THE GENERAL POSITIONING ACCURACY OF ZHUHAI-1 WITH THE CONTROLLING OF GCPS (UNIT: METER).

Item		X direction	Y direction	Planar
Absolute positioning accuracy	Max error	111.60	59.30	123.65
	Min error	91.57	50.37	107.72
	RMSE	**103.86**	**55.95**	**117.97**
	Variance	26.59	7.23	12.65
Relative positioning accuracy	Max error	12.16	5.51	12.19
	Min error	0.50	0.01	0.63
	RMSE	**5.12**	**2.59**	**5.74**
	Variance	6.60	1.61	6.83
General positioning accuracy of Zhuhai-1	Max error	33.45	18.74	34.59
	Min error	0.03	0.92	7.51
	RMSE	**12.41**	**9.50**	**15.63**
	Variance	85.78	30.56	59.57

We statistically analyzed the max error, min error and calculate the RMSE and variance of all estimated points in x direction, y direction and plane, respectively. The estimated trajectory and ground truth of the fishing boat are basically in the same area with an APE of 117.97 m in this experiment. It can be seen that there are some systematic errors between the estimated trajectory and ground truth, resulting in an overall shift in absolute positions. We have a good relative positioning result with an RPE of 5.74 m, as shown in Table III and Fig. 10. After eliminating the influence of absolute positioning, an overlap of the normalized estimated trajectory and normalized ground truth is shown in Fig.10(c). And CE90 (Circular Error at 90th percentile) is also selected for evaluating the relative positioning accuracy, as shown in Fig.10(d). CE90 is the circular error at the 90th percentile of all position errors and is a widely adopted error metric for geolocation accuracy in remote sensing. The CE90 of our RPE is 8.6 m, which means that a minimum of 90 percent of the points has an RPE less than the stated value.

And we have a better result in the estimation of speed. The RMSE of the estimated speed is 0.34 m/s (1.22 km/h) with a variance of 0.003, while the boat moved at a mean speed of 4.42 m/s (15.91 km/h). And the accuracy of the estimated speed is 92.31 %, calculated by (14). The direction of the motion is estimated with an RMSE of 0.02 rad (1.15 $^\circ$) and a variance of 7.54×10^{-5}. As shown in Fig. 9, the fishing boat moved with a approximate uniform linear pattern, leading to a very small variance in ground truth and estimated values.

We can not compare the results directly with the aforementioned conventional method, because the establishment of rigid geometric imaging model needs some parameters which are not feasible to us at present. However, we are provided with the general positioning accuracy of Zhuhai-1 video satellite, which is evaluated using conventional method with the controlling of Ground Control Points(GCP), as shown in Table III. We can evaluate our results indirectly to some extent. With the controlling of GCPs, the absolute positioning accuracy of Zhuhai-1 generally reaches about 15.63 m. However, we do not know the accuracy without

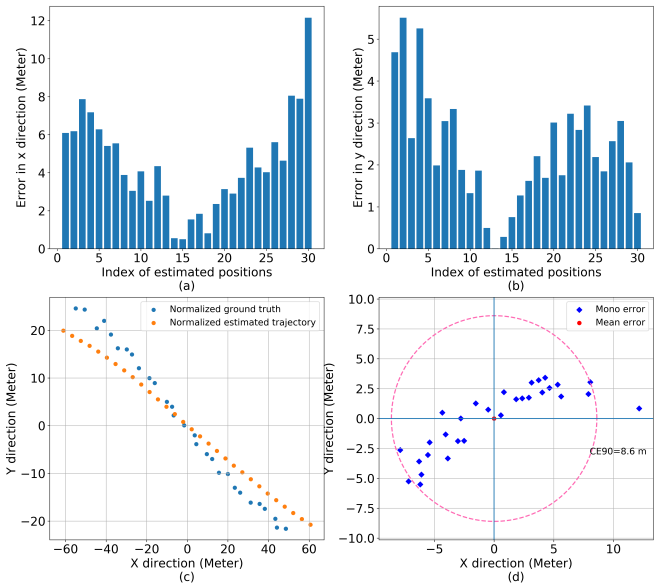

Fig. 10. Analyses for RPE of the fishing boat. (a) The RPE along the x direction. (b) The RPE along the y direction. (c) The comparison of the normalized ground truth and normalized estimated trajectory. (d) The distribution of RPE calculated by CE90. The pink circle is the CE90 of 8.6 m, which means that 90 percent of the points are less than it.

GCPs, which is usually larger than it. For our method, we achieve a reasonable result for positioning without the help of GCPs, which is more general in application and efficient in processing. And compared with the provided general positioning accuracy, we achieve a smaller variance and a more stable performance. Last but not least, geo-positioning is finished in near real-time rather than post-processing like conventional methods. In the following work, we will use other video satellites(such as JiLin-1, SkySat-2) for a thorough comparison.

V. DISCUSSION

We propose a tracking and positioning framework for ground moving targets in satellite videos leverage on SLAM and achieve fair results, while there is still room to improve. We first analyze factors influencing the positioning accuracy, then discuss open issues and related research directions.

Due to the complexity of the entire process, the positioning accuracy is affected by many factors, which we summarize in three aspects: pose estimation, target tracking, triangulation and transformation. Pose estimation is mainly affected by the accuracy of sensor measurements, the visual odometry of SLAM, and different fusion strategies. And target tracking is affected by different size, shape and moving pattern of moving objects. The accuracy of triangulation is affected by the estimated pose and feature matching. Finally, the accuracy of parameters used in transformation, such as EOP parameters, camera intrinsic, also play a important role for final accuracy.

To facilitate related follow-up research, we list some open problems and potential directions for this topic.

1) The evaluation and fusion of sensor and visual odometry data. How to eliminate the error in measurements is

meaningful. And how to select the proper interval of frames for pose estimation is also critical. Finally, the evaluation of the accuracy and uncertainty of data and fusion with different weights according to reliability is also a problem.

2) Tracking for different moving targets. We track large single objects in this paper. While there may be two research directions: a accurate instance-level tracking method for multiple targets considering the efficiency, a general method for targets with different sizes and shapes.

3) Global optimization. Due to the unique one-way motion of satellites, we do not find a visual loop closure conventionally adopted in SLAM, resulting in the lack of global optimization in this paper. However, it is important for eliminating systematic and accumulative errors.

4) Consideration practical environment of satellites. Some simplifications are made for research in this paper. We think the satellite runs stably at a smooth orbit, ignoring complex factors (such as the non-spherical gravitational field of Earth) that influence satellite motion, and high-frequency jitter. And we use the mean radius of Earth and ignore the earth rotation in a short time. Meanwhile, we make the assumption that frames are taken in the same plane on the sea level. However, it does not hold in some cases, leading a decline of positioning accuracy. It is necessary to refine these simplifications and assumptions for a better result.

5) Analysis on operational domain. It is also important to clarify the operational domain in real applications. And a better simulation environment for experiments is also needed for convenient experiments.

We are currently exploring ideas in the aforementioned directions for a better positioning accuracy and framework.

VI. Conclusion

In this paper, we propose a tracking and positioning framework for ground moving targets in satellite videos. After some experiments, our framework is proved to be feasible, and reasonable results have been achieved. Compared with other related research, our framework can run in near-real-time and positioning moving objects in the world coordinate (such as WGS84), which greatly improves the value and application potential of satellite videos. However, due to the complicity of the entire process, we present initial ideas and preliminary version of the framework in this paper, and there are still some open issues need to be solved. Finally, potential directions and planed work are discussed.

References

[1] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "Orb-slam: a versatile and accurate monocular slam system," *IEEE transactions on robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[2] L. Zhao and P. Chen, "Moving target autonomous positioning based on vision for uav," in *China Satellite Navigation Conference (CSNC) 2015 Proceedings: Volume III*. Springer, 2015, pp. 691–701.

[3] J. Nielsen and R. Beard, "Ground target tracking using a monocular camera and imu in a nonlinear observer slam framework," in *2018 Annual American Control Conference (ACC)*. IEEE, 2018, pp. 6457–6462.

[4] J. H. Choi, D. Lee, and H. Bang, "Tracking an unknown moving target from uav: Extracting and localizing an moving target with vision sensor based on optical flow," in *The 5th International Conference on Automation, Robotics and Applications*. IEEE, 2011, pp. 384–389.

[5] P. Chen, Y. Dang, R. Liang, W. Zhu, and X. He, "Real-time object tracking on a drone with multi-inertial sensing data," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 1, pp. 131–139, 2017.

[6] S. Chen, S. Guo, and Y. Li, "Real-time tracking a ground moving target in complex indoor and outdoor environments with uav," in *2016 IEEE International Conference on Information and Automation (ICIA)*. IEEE, 2016, pp. 362–367.

[7] Y.-T. Wang, Y.-C. Feng, and D.-Y. Hung, "Detection and tracking of moving objects in slam using vision sensors," in *2011 IEEE International Instrumentation and Measurement Technology Conference*. IEEE, 2011, pp. 1–5.

[8] J. Min, J. Kim, H. Kim, K. Kwak, and I. S. Kweon, "Hybrid vision-based slam coupled with moving object tracking," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 867–874.

[9] T.-D. Vu, O. Aycard, and N. Appenrodt, "Online localization and mapping with moving object tracking in dynamic outdoor environments," in *2007 IEEE Intelligent Vehicles Symposium*. IEEE, 2007, pp. 190–195.

[10] Q. Baig, T.-D. Vu, and O. Aycard, "Online localization and mapping with moving objects detection in dynamic outdoor environments," in *2009 IEEE 5th International Conference on Intelligent Computer Communication and Processing*. IEEE, 2009, pp. 401–408.

[11] T. Yang, X. Wang, B. Yao, J. Li, Y. Zhang, Z. He, and W. Duan, "Small moving vehicle detection in a satellite video of an urban area," *Sensors*, vol. 16, no. 9, p. 1528, 2016.

[12] B. Du, Y. Sun, S. Cai, C. Wu, and Q. Du, "Object tracking in satellite videos by fusing the kernel correlation filter and the three-frame-difference algorithm," *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 2, pp. 168–172, 2017.

[13] H. Li and Y. Man, "Moving ship detection based on visual saliency for video satellite," in *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. IEEE, 2016, pp. 1248–1250.

[14] B. Du, S. Cai, and C. Wu, "Object tracking in satellite videos based on a multiframe optical flow tracker," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 12, no. 8, pp. 3043–3055, 2019.

[15] S. Xuan, S. Li, M. Han, X. Wan, and G.-S. Xia, "Object tracking in satellite videos by improved correlation filters with motion estimations," *IEEE Transactions on Geoscience and Remote Sensing*, 2019.

[16] Y. Guo, D. Yang, and Z. Chen, "Object tracking on satellite videos: A correlation filter-based tracking method with trajectory correction by kalman filter," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 12, no. 9, pp. 3538–3551, 2019.

[17] X. WANG, F. LI, L. XIN, J. MA, X. YANG, and X. CHANG, "Moving targets detection for satellite-based surveillance video," in *IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 2019, pp. 5492–5495.

[18] G. Kopsiaftis and K. Karantzalos, "Vehicle detection and traffic density monitoring from very high resolution satellite video data," in *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. IEEE, 2015, pp. 1881–1884.

[19] S. J. Kelly, "A monocular slam method to estimate relative pose during satellite proximity operations," 2015.

[20] D. Thomas, S. Kelly, and J. Black, "A monocular slam method for satellite proximity operations," in *2016 American Control Conference (ACC)*. IEEE, 2016, pp. 4035–4040.

[21] F. Andert, N. Ammann, and B. Maass, "Lidar-aided camera feature tracking and visual slam for spacecraft low-orbit navigation and planetary landing," in *Advances in Aerospace Guidance, Navigation and Control*. Springer, 2015, pp. 605–623.

[22] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the seventh IEEE international conference on computer vision*, vol. 2. Ieee, 1999, pp. 1150–1157.

[23] X. Chen, B. Zhang, M. Cen, H. Guo, T. Zhang, and C. Zhao, "Srtm dem-aided mapping satellite-1 image geopositioning without ground control points," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 11, pp. 2137–2141, 2017.

[24] Z. Zhang and D. Scaramuzza, "A tutorial on quantitative trajectory evaluation for visual (-inertial) odometry," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 7244–7251.