SCIENCE CHINA
Information Sciences

• **Supplementary File** •

An uncalibrated downward-looking visual compass for UAVs based on clustered point features

Ying LIU[1], Yu ZHANG[1*], Ping LI[1] & Bin XU[2]

[1] *State Key Laboratory of Industrial Control Technology, College of Control Science and Engineering, Zhejiang University, Hangzhou* 310027*, China;*
[2] *School of Automation, Northwestern Polytechnical University, Xi'an* 710072*, China*

Appendix A Algorithm 1

Algorithm A1 Building point features from density-based cluster

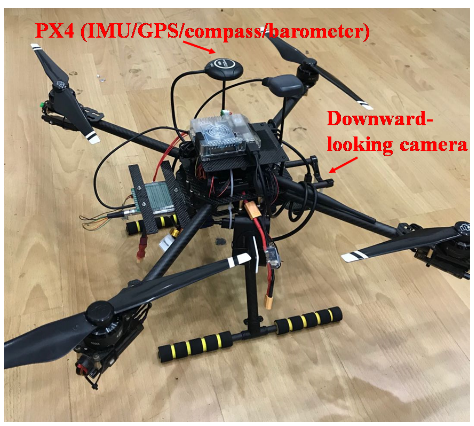
Require: The raw image in current frame I;
1: **if** I is the first frame **then**
2: Find all m ORB keypoints to form a sample set: $\boldsymbol{D}_K^c = \{k_1^c, k_2^c, \cdots, k_m^c\}$;
3: Implement DBSCAN in $\boldsymbol{D}_K^c$ to divide m keypoints into n clusters $\boldsymbol{C}_1^c, \boldsymbol{C}_2^c, \cdots, \boldsymbol{C}_n^c$, and ignore the noises;
4: **for** each $\boldsymbol{C}_i^c = k_{i1}^c, k_{i2}^c, \cdots, k_{im_i}^c, i = 1, 2, \cdots, n$ **do**
5: Calculate c_i^c according to Eq. (2);
6: **end for**
7: Find ORB descriptor for $\boldsymbol{D}_C^c = \{c_1^c, c_2^c, , c_n^c\}$;
8: Set I to be the keyframe;
9: **else**
10: do 3-8 steps;
11: Match $\boldsymbol{D}_C^c$ and $\boldsymbol{D}_C^k$ by Brute-Force matching;
12: **for** each matched $c_i^c \in \boldsymbol{D}_C^c$ and $c_i^k \in \boldsymbol{D}_C^k$ **do**
13: **if** $N_{cp}^c - N_{cp}^k \leqslant T_{\Delta cs}$ and $distance(c_i^c, c_j^k) \leqslant T_{\Delta cd}$ **then**
14: $c_i^c = c_i^{c*} \in \boldsymbol{D}_C^{c*}$ and $c_i^k = c_i^{k*} \in \boldsymbol{D}_C^{k*}$;
15: **end if**
16: **end for**
17: **end if**
18: **return** $\boldsymbol{D}_C^{c*} = \{c_1^{c*}, c_2^{c*}, \cdots, c_{N_c}^{c*}\}$ and $\boldsymbol{D}_C^{k*} = \{c_1^{k*}, c_2^{k*}, \cdots, c_{N_c}^{k*}\}$

Appendix B Experiments

In these experiments, we compared three algorithm in four cases:
(1) Static UAV in static environment;
(2) Static UAV in dynamic environment;
(3) Dynamic UAV in static environment;
(4) Dynamic UAV in dynamic environment;

The position of the UAV was intended to be stable and the attitude of the camera was equal to the UAV since the camera was mounted on the UAV. The static UAV means the yaw angle of it was kept as stable as possible in position model. Dynamic UAV means the UAV is only rotated in the heading angle. In static environment, the scene watched by the camera is stable while there are some moving objects in dynamic environment. We did several experiments of each cases in different locations in our campus. And we did not use extra sensor to offer Ψ_1, so Ψ_1 was equal to 0 degree in our experiments. The comparison results are shown in Figure B2.

* Corresponding author (email: zhangyu80@zju.edu.cn)



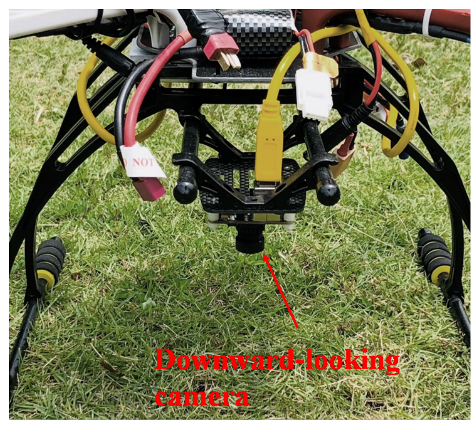
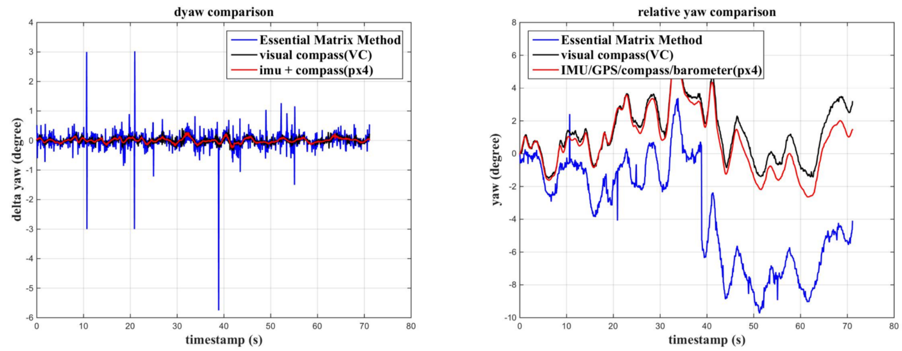
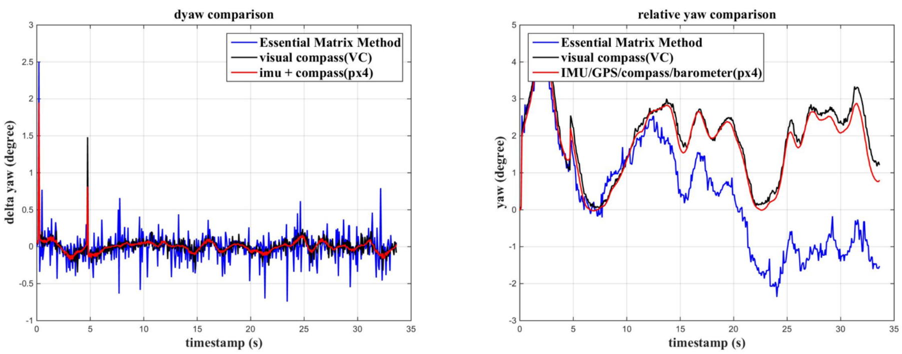


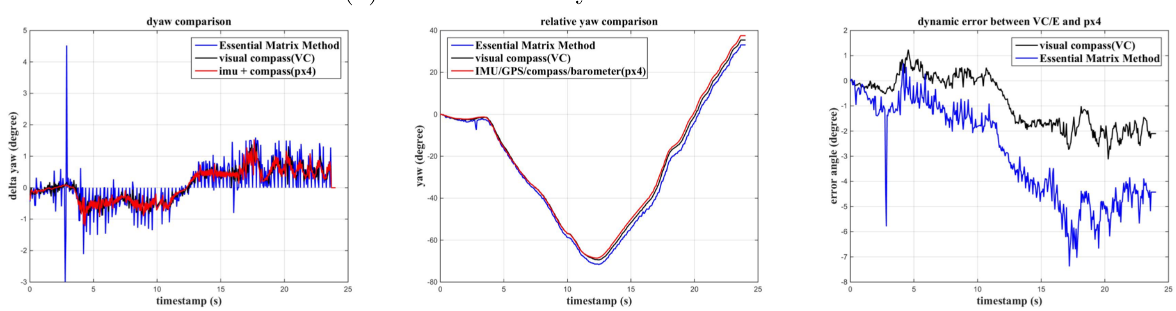
Figure B1 The UAV used in the outdoor experiments, which was implemented a flight controller PX4 and a downward-looking monocular camera based on DJI M100. (a) The UAV platform; (b) The camera mounted on the UAV.



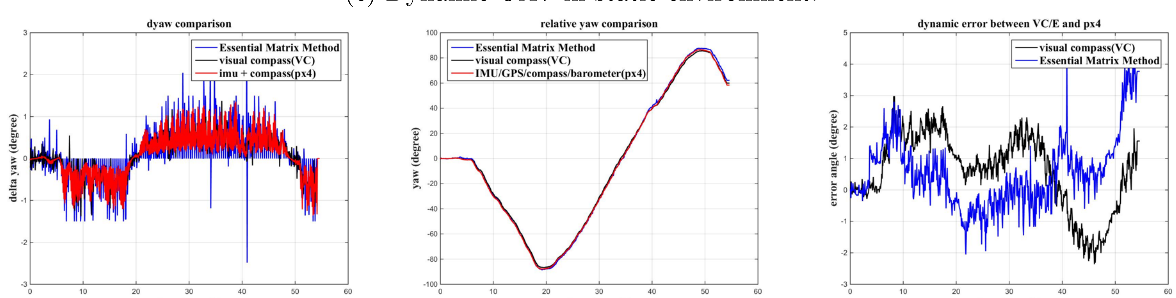
(a) Static UAV in static environment.



(b) Static UAV in dynamic environment.



(c) Dynamic UAV in static environment.



(d) Dynamic UAV in dynamic environment

Figure B2 (a) and (b) show the performances of these three algorithms with static UAV in static and dynamic environment respectively. The left figure shows the comparison of the yaw angle increment between two adjacent frames, denoted as $\Delta\Psi_{Ad}$. The right figure shows the comparison of the final output Ψ_c^1. (c) and (d) show the performances of these three algorithms with dynamic UAV in static and dynamic environment respectively. The figures on the left and in the middle show the comparison of $\Delta\Psi_{Ad}$ and Ψ_c^1 accordingly. And the right picture displays the error of Ψ_c^1 between the final output of these three algorithms.