Proceedings of the 34th Chinese Control Conference
July 28-30, 2015, Hangzhou, China

A Mismatching Eliminating Method Based on Camera Motion Information

Li Chuang[1], Dong Hongxin[1], Quan Quan[1]

1. School of Automation Science and Electrical Engineering, Beihang University, Beijing 100191, P. R. China

Chuangcg@126.com;dhx_buaa@buaa.edu.cn;qq_buaa@buaa.edu.cn

Abstract: Most visual navigation relies on point matching and many of the matching point pairs are faulty. Generally, they are eliminated by the Random Sample Consensus (RANSAC) algorithms. However, RANSAC algorithms are often time-consuming and their parameters are hard to adjust. For such a purpose, a new method is proposed to eliminate the error matching points of every two consecutive images based on motion information. The proposed method is applicable to the cases that the motion information is available and the interest points are all from a horizontal plane. It does not require iteration and thus time-efficient. To demonstrate the effectiveness, the proposed method is tested on synthetic data of both simulation points and real images. The results show that the proposed method has better performance over RANSAC algorithms.

Key Words: matching; velocity; visual navigation; RANSAC; quadrotor

1 Introduction

Unmanned Aerial Vehicle (UAV) navigation is of great importance to many fields nowadays, such as automatic flight [1] and automatic irrigate [2]. A common navigation way is the use of Global Position System (GPS) [3], but it is not available in many cases like indoor environments or experiment close to buildings. Another navigation way is visual navigation. Because of the advantages like small, light, low power demanding and high-accuracy character, visual navigation is applied by more and more engineers and researchers to short-distance and high-accuracy UAV navigation [4].

In visual navigation, the most common way is to find the matching points, and then obtain position and attitude information through the multiple view geometry. However, it is not easy to find correct matching points [5]. Often, there exists a great part of error matching point pairs, and the error matching pairs will further have a negative effect on the following work. For such a purpose, a method and its variants, called Random Sample Consensus (RANSAC) algorithms, are widely used to eliminate the error matching. However, RANSAC algorithms are often time-consuming and their parameters are hard to adjust [6]. These motivate us to propose a time-efficient and easy-to-use method to eliminate the common error matching points.

The proposed method aims to simplify the eliminating process through utilizing the camera's motion information, under the assumption that all the interest points are on a horizontal plane. The main idea is to utilize camera's motion information to calculate the positions in sense of probability where the interest points should be in the next image. Then, the matching possibilities of the given matching point pairs are measured. The matching pairs with high matching probability will be reserved, and others will be eliminated. The resulting point matching will in turn help to refine the camera's motion information. The experiment result shows the effectiveness of the proposed method, and it is applicable to many cases. For example, a facing down camera is mounted on an UAV to take pictures of the ground.

The paper is organized as follows. Section 2 describes some preliminaries including the multi-dimensional normal distribution and the pinhole camera model. Section 3 makes a detailed description of the problem formulation. In Section 4, the detailed eliminating procedures are described. Simulations are given in Section 5.

In this paper, the following notation is used. $\mathbb{R}^n$ denotes n-dimensional Euclidean space. ${}^b\mathbf{R}_a$ and ${}^b\mathbf{T}_a$ denotes a rotation matrix and a translation matrix from coordinate $\{a\}$ to coordinate $\{b\}$, respectively. $\mathbf{e}_3$ denotes a vector $[0\ 0\ 1]^T$. If $\mathbf{A}$ is a matrix, $\mathbf{A}^T$ denotes its transposition. If $\mathbf{x}$ and $\mathbf{y}$ are vectors of the same dimension, $\mathbf{x} \times \mathbf{y}$ denotes the cross multiply of $\mathbf{x}$ and $\mathbf{y}$. $\mathbf{X}^{-1}$ denotes the inverse matrix of reversible square matrix $\mathbf{X}$. $|\cdot|$ denotes the absolute value of a real number. $\mathbf{0}_{m\times n}$ denotes a zero vector or a zero matrix with dimension $m \times n$. $\mathbf{I}_n$ is the identity matrix of order n.

2 Preliminaries

2.1 Multi-dimensional Normal Distribution

Let $\mu \in \mathbb{R}^n$ and $\mathbf{\Sigma}$ be a n-dimensional real symmetric positive definite matrix. If the joint probability density function of $\mathbf{X}$ is

$$f(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^n \det(\mathbf{\Sigma})}} e^{-\frac{1}{2}(\mathbf{x}-\mu)^T \mathbf{\Sigma}^{-1} (\mathbf{x}-\mu)}, \quad (1)$$

then $\mathbf{X}$ obeys a gaussian distributions with mean μ and variance $\mathbf{\Sigma}$, denoted by

$$\mathbf{X} \sim \mathcal{N}(\mu, \mathbf{\Sigma}).$$

If a n-dimensional random vector $\mathbf{X} \sim \mathcal{N}(\mu, \mathbf{\Sigma})$, then

$$\mathbf{Y} = \mathbf{A}\mathbf{X} + \mathbf{b} \sim \mathcal{N}(\mathbf{A}\mu + \mathbf{b}, \mathbf{A}\mathbf{\Sigma}\mathbf{A}^T), \quad (2)$$

where and $\mathbf{A} \in \mathbb{R}^{n\times n}$ and $\mathbf{b} \in \mathbb{R}^n$ [7].

2.2 Camera Model

A linear pinhole camera model performs the transformation from $\mathbf{p}$ in $\{w\}$ to the camera coordinate system $\{c\}$, then to the image point $\mathbf{m}$ in ${}^iO\text{-}{}^iX{}^iY{}^iZ$, as shown in Fig.1. [8] The vectors ${}^w\mathbf{p} = [{}^wX\ {}^wY\ {}^wZ]^T$ and ${}^c\mathbf{p} = [{}^cX\ {}^cY$

This work was supported by National Natural Science Foundation of China (No. 61473012,51375462).

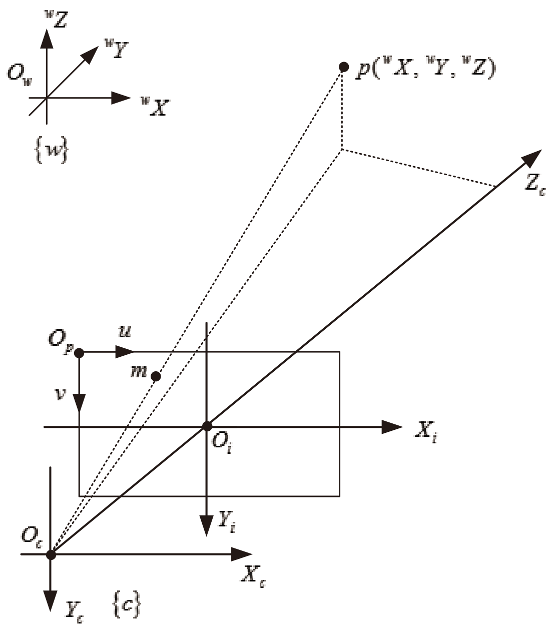


Fig. 1: Imaging Model

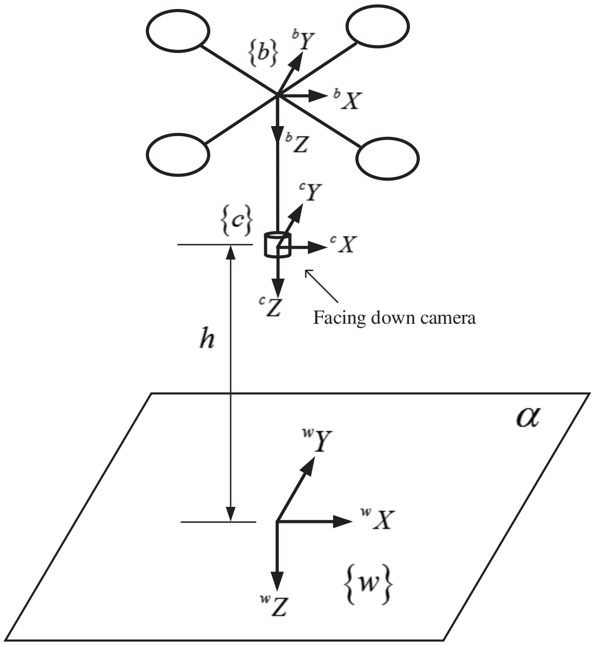


Fig. 2: Coordinate Model

${}^cZ]^T$ denote the coordinates of $\mathbf{p}$ in $\{w\}$ and $\{c\}$, respectively. They satisfy

$$ {}^c\mathbf{p} = {}^c\mathbf{R}_w \, {}^w\mathbf{p} + {}^c\mathbf{T}_w, $$

where $({}^cR_w, {}^cT_w)$ is the transformation matrix from $\{w\}$ to $\{c\}$. Furthermore, the relation between a spatial point ${}^w\mathbf{p}$ and its corresponding normal image point $[x_n \; y_n]^T$ is written as

$$ \lambda \begin{bmatrix} x_n \\ y_n \\ 1 \end{bmatrix} = {}^c\mathbf{R}_w \, {}^w\mathbf{p} + {}^c\mathbf{T}_w, $$

where $\lambda = {}^cz, x_n = \frac{{}^cX}{{}^cZ}, y_n = \frac{{}^cY}{{}^cZ}$.

The real image point $[u \; v]^T$ is obtained from

$$ \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{K} \begin{bmatrix} x_n \\ y_n \\ 1 \end{bmatrix}, \tag{3} $$

and

$$ \mathbf{K} = \begin{bmatrix} \alpha_x & 0 & u_0 \\ 0 & \alpha_y & v_0 \\ 0 & 0 & 1 \end{bmatrix}, $$

where α_x and α_y are the scale factors in image u and v axes; $[u_0, v_0]^T$ are the coordinate of the principal point.

3 Problem Formulation

The used coordinate systems are shown in Fig.2, where $\{w\}$ denotes the world coordinate, $\{c\}$ denotes the camera coordinate and $\{b\}$ denotes the quadrotor's body coordinate. The facing down camera is mounted on the quadrotor. The variable α denotes the horizontal plane and h denotes the distance between the plane α and the center of the camera, denoted by

$$ h = {}^c\mathbf{T}_w(3), $$

where ${}^c\mathbf{T}_w(3)$ is the third element of ${}^c\mathbf{T}_w$.

Some assumptions are made as follows.

Assumption 1. All the interest points ${}^w\mathbf{p}_k \in \alpha = \left\{ {}^w\mathbf{p} | {}^wZ = 0, {}^w\mathbf{p} \in \mathbb{R}^3 \right\}, k = 1, \cdots, N$.

Assumption 2. The variables $h, {}^c\mathbf{R}_w, \mathbf{K}$ and $\mathbf{v}^m, \mathbf{w}^m$ are measurable. The relation between measured value $\mathbf{v}^m$ and $\mathbf{w}^m$ and true value $\mathbf{v}$ and $\mathbf{w}$ is

$$ \begin{aligned} \mathbf{v} &= \mathbf{v}^m + \varepsilon_{\mathbf{v}} \\ \mathbf{w} &= \mathbf{w}^m + \varepsilon_{\mathbf{w}} \end{aligned} $$

where

$$ \varepsilon_{\mathbf{v}} \sim \mathcal{N}\left(\mathbf{0}_{3\times1}, \sigma_{\mathbf{v}}^2 \mathbf{I}_3\right), \varepsilon_{\mathbf{w}} \sim \mathcal{N}(\mathbf{0}_{3\times1}, \sigma_{\mathbf{w}}^2 \mathbf{I}_3). $$

Assumption 3. At the time $t + \Delta t$, the pairs $(\mathbf{p}_i(t), \mathbf{q}_i(t + \Delta t))$ are initial matching point pairs, namely there exists outliers, where $\mathbf{p}_i(t) \in S_p \subset \mathbb{R}^2, \mathbf{q}_i(t + \Delta t) \in S_q \subset \mathbb{R}^2, i = 1, \cdots, N$.

Assumption 4. The sampling time Δt is sufficiently small that $\mathbf{p}(t + \Delta t) = \mathbf{p}(t) + \Delta t \cdot \dot{\mathbf{p}}(t)$ holds approximately.

Under *Assumptions 1-4*, the objective is to eliminate the error matching point pairs in $(\mathbf{p}_i(t), \mathbf{q}_i(t + \Delta t))$ based on the camera's motion information, $i = 1, \cdots, N$.

Remark 1. In practice, there are many cases in which the object is a plane or can be considered as a plane. When a facing down camera mounted on a drone flying in open indoor environment, the camera scene is a plane. In other cases, such as outdoor flight, drones usually fly at a high altitude, where the camera scene can be considered approximately as a plane.

Remark 2. The height h can be obtained through the altitude sensor, such as barometric altimeter, radio altimeter or ultrasonic barometer in different condition with acceptable precision; ${}^c\mathbf{R}_w$ and $\mathbf{w}^m$ can be acquired from the IMU; the intrinsic parameter of the camera $\mathbf{K}$ is available by the calibration offline. The velocity $\mathbf{v}^m$ can be generated by existing methods [9],[10],[11]. It is pointed out that the resulting point matching will in turn help to refine the camera's velocity.

Remark 3. Moving with $\mathbf{v}$ and $\mathbf{w}$, the camera takes two consecutive pictures of the plane α in a sampling time Δt. The interest points can be extracted from them. Then by image matching algorithms, N pairs of matching point pairs $(\mathbf{p}_i\,(t)\,,\mathbf{q}_i\,(t+\Delta t))$, $(i=1,2,...,N)$ are obtained in the two point sets S_p and S_q.

Next, the proposed method will be introduced in the Section 4 in detail. For simplicity, this paper will focus on one pair of the matching point pairs, namely $(\mathbf{p}\,(t)\,,\mathbf{q}\,(t+\Delta t))$.

4 Main Result

The derivative of the point $\mathbf{p}\,(t)$, namely $\dot{\mathbf{p}}\,(t)$, can be calculated based on $\mathbf{v}^m,\mathbf{w}^m,{}^c\mathbf{R}_w$ and h. Then, the next position of the point, namely $\mathbf{p}\,(t+\Delta t)$, can be obtained through $\dot{\mathbf{p}}\,(t)$ and $\mathbf{p}\,(t)$. But the measurement $\mathbf{v}^m,\mathbf{w}^m$ are subject to noise. The probability density distribution of the point in the next image is derived. Finally, a threshold is chosen to eliminate the error matching. The three steps of the proposed method mentioned above will be introduced in detail in Section 4.1, Section 4.2 and Section 4.3, respectively. The concrete procedure will be described in Section 4.4. In the following, for convenience, the variable t will be omitted except when necessary.

4.1 Computing of $\dot{\mathbf{p}}$

A 3D point ${}^c\mathbf{p}=[{}^cX\ {}^cY\ {}^cZ]^T$ in the camera frame is projected to a 2D point with coordinates $\mathbf{p}=[x\ y]^T$. By recalling Section 2.2, it holds

$$\left\{\begin{array}{l} x=\frac{{}^cX}{{}^cZ}=\frac{u-u_0}{\alpha_x}\\ y=\frac{{}^cY}{{}^cZ}=\frac{v-v_0}{\alpha_y}\end{array}\right.,$$

where $\mathbf{m}=[u\ v]^T$ is the coordinates of the point in pixel frame, and $(u_0,v_0,\alpha_x,\alpha_y)$ are the camera intrinsic parameters.

Taking the derivative of the equations above results in

$$\left\{\begin{array}{l} \dot{x}=\frac{{}^c\dot{X}}{{}^cZ}-\frac{{}^cX{}^c\dot{Z}}{{}^cZ^2}=\frac{{}^cX-x\,{}^c\dot{Z}}{{}^cZ}\\ \dot{y}=\frac{{}^c\dot{Y}}{{}^cZ}-\frac{{}^cY\,{}^c\dot{Z}}{{}^cZ^2}=\frac{(\,{}^cY-y\,{}^c\dot{Z})}{{}^cZ}\end{array}\right.. \tag{4}$$

Relation between the velocity of ${}^c\mathbf{p}$ and the spatial velocity of camera is

$${}^c\dot{\mathbf{p}}=-\mathbf{v}-\mathbf{w}\times\ {}^c\mathbf{p},$$

which can be written as

$$\left\{\begin{array}{l} {}^c\dot{X}=-v_x-w_y\ {}^cZ+w_z\ {}^cY\\ {}^c\dot{Y}=-v_y-w_z\ {}^cX+w_x\ {}^cZ\\ {}^c\dot{Z}=-v_z-w_x\ {}^cY+w_y\ {}^cX\end{array}\right.. \tag{5}$$

Substituting (5) into (4) results in

$$\dot{\mathbf{p}}=\mathbf{A}_{x,y}\mathbf{v}+\mathbf{B}_{x,y}\mathbf{w}, \tag{6}$$

where

$$\begin{aligned}\mathbf{A}_{x,y}&=\frac{1}{Z}\left[\begin{array}{ccc}-1&0&x\\0&-1&y\end{array}\right],\\ \mathbf{B}_{x,y}&=\left[\begin{array}{ccc}xy&-(1+x^2)&y\\(1+y^2)&-xy&-x\end{array}\right]\end{aligned}$$

and the depth of the point at (x,y) is

$$Z=\frac{h}{\mathbf{e}_3^T\ {}^c\mathbf{R}_w[x,y,1]^T}. \tag{7}$$

4.2 Probability Density Distribution

In theory, according to *Assumption 4*, the exact position of the point $\mathbf{p}$ in the next image can be calculated through

$$\mathbf{p}\,(t+\Delta t)=\mathbf{p}\,(t)+\Delta t\cdot\dot{\mathbf{p}}\,(t) \tag{8}$$

where $\dot{\mathbf{p}}\,(t)$ is obtained by (6). However, the real $\mathbf{v}$ and $\mathbf{w}$ cannot be obtained. Only noisy $\mathbf{v}^m$ and $\mathbf{w}^m$ are available. Consequently, by using $\mathbf{v}^m$ and $\mathbf{w}^m$, the position of the point $\mathbf{p}$ in the next image, namely $\mathbf{p}\,(t+\Delta t)$, is uncertain, which obeys a probability density distribution.

According to *Assumption 2*, the equation (6) is rewritten as

$$\dot{\mathbf{p}}=\mathbf{A}_{x,y}(\mathbf{v}^m+\varepsilon_{\mathbf{v}})+\mathbf{B}_{x,y}(\mathbf{w}^m+\varepsilon_{\mathbf{w}}).$$

So, according to *Assumption 4*, $\mathbf{p}(t+\Delta t)$ can be written as

$$\mathbf{p}(t+\Delta t)=\mathbf{A}_{x,y}\varepsilon_{\mathbf{v}}+\mathbf{B}_{x,y}\varepsilon_{\mathbf{w}}+\mu_p, \tag{9}$$

in which

$$\mu_p=\mathbf{A}_{x,y}\mathbf{v}^m+\mathbf{B}_{x,y}\mathbf{w}^m+\mathbf{p}.$$

According to the equation (2) in preliminaries, the distribution of $\mathbf{p}(t+\Delta t)$ is

$$\mathbf{p}(t+\Delta t)\sim\mathcal{N}(\mu_p,\mathbf{\Sigma}_p),$$

in which

$$\mathbf{\Sigma}_p=\sigma_{\mathbf{v}}^2\mathbf{A}_{x,y}\mathbf{A}_{x,y}^T+\sigma_{\mathbf{w}}^2\mathbf{B}_{x,y}\mathbf{B}_{x,y}^T. \tag{10}$$

4.3 Theory on Error Matching Eliminating

As described above, $\mathbf{p}\,(t+\Delta t)$ is the true value of the point at position $\mathbf{p}\,(t)$ in the next image. The relation between them is given in (9). In the following, the matching probability of the pair $(\mathbf{p}\,(t)\,,\mathbf{q}\,(t+\Delta t))$ is investigated, namely the distance between $\mathbf{p}\,(t+\Delta t)$ and $\mathbf{q}\,(t+\Delta t)$ in sense of probability. To proceed further, a proposition is proposed first.

Proposition 1. For n-dimension $\mathbf{X}\sim\mathcal{N}(\mu,\mathbf{\Sigma})$, it holds $\mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{U}^T\mathbf{X}\sim\mathcal{N}(\mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{U}^T\mu,\mathbf{I}_n)$, where $\mathbf{\Sigma}=\mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$, $\mathbf{U}\in\mathbb{R}^{n\times n}$ is a rotation matrix, and $\mathbf{\Lambda}\in\mathbb{R}^{n\times n}$ is a diagonal matrix.

Proof. The conclusion can be derived by

$$\begin{aligned}\mathrm{E}\left(\mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{U}^T\mathbf{X}\right)&=\mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{U}^T\mu\\ \mathrm{cov}\left(\mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{U}^T\mathbf{X}\right)&=\mathrm{cov}\left(\mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{U}^T\mathbf{X}\mathbf{X}^T\mathbf{U}\mathbf{\Lambda}^{-\frac{1}{2}}\right)\\ &=\mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{U}^T\mathrm{cov}\left(\mathbf{X}\mathbf{X}^T\right)\mathbf{U}\mathbf{\Lambda}^{-\frac{1}{2}}\\ &=\mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{U}^T\mathbf{\Sigma}\mathbf{U}\mathbf{\Lambda}^{-\frac{1}{2}}\\ &=\mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{U}^T\mathbf{U}\mathbf{\Lambda}\mathbf{U}^T\mathbf{U}\mathbf{\Lambda}^{-\frac{1}{2}}\\ &=\mathbf{I}_n.\end{aligned}$$

□

The distance between $\mathbf{p}\,(t+\Delta t)$ and $\mathbf{q}\,(t+\Delta t)$, defined as $\mathbf{d}=\mathbf{p}(t+\Delta t)-\mathbf{q}\,(t+\Delta t)$, can be written as

$$\mathbf{d}=\mathbf{p}(t+\Delta t)-\mu_p+\mu_p-\mathbf{q}\,(t+\Delta t)\,.$$

Its mean and covariance are

$$\begin{aligned} \mathrm{E}(\mathbf{d}) &= \mathrm{E}(\mathbf{p}(t+\Delta t)-\mu_p)+\mathrm{E}(\mu_p-\mathbf{q}(t+\Delta t)) \\ &= \mu_p-\mathbf{q}(t+\Delta t) \end{aligned}$$

and

$$\begin{aligned} \mathrm{Cov}(\mathbf{d}) &= \mathrm{Cov}(\mathbf{p}(t+\Delta t))+\mathrm{Cov}(\mathbf{q}(t+\Delta t)) \\ &= \mathbf{\Sigma}_p. \end{aligned}$$

Therefore, one has

$$\mathbf{d} \sim \mathcal{N}(\mu_p-\mathbf{q}(t+\Delta t), \mathbf{\Sigma}_p).$$

The distance should be normalized because the moving distances of the points during the same time interval are related to the positions in an image. For example, as shown in Fig.3, for a pure yaw rotation, the moving distances of the points at the center are small, whereas those will be large at the edge. Let $\mathbf{\Sigma}_p = \mathbf{U}_p\mathbf{\Lambda}_p\mathbf{U}_p^T$, where $\mathbf{U}_p \in \mathbb{R}^{2\times 2}$ is a rotation matrix, and $\mathbf{\Lambda}_p \in \mathbb{R}^{2\times 2}$ is a diagonal matrix. According to *Proposition 1*, the distance $\mathbf{d}$ is modified as

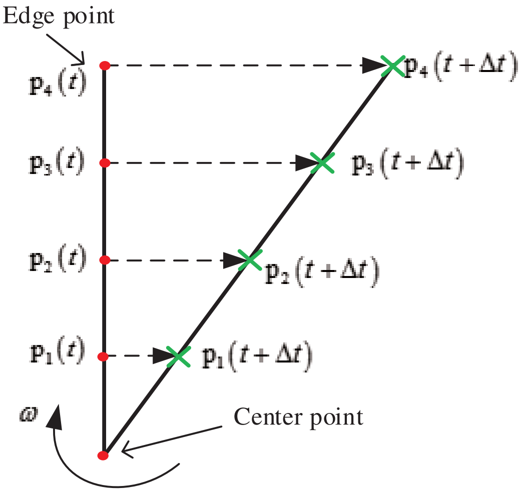

Edge point
$\mathbf{p}_4(t)$
$\mathbf{p}_4(t+\Delta t)$
$\mathbf{p}_3(t)$
$\mathbf{p}_3(t+\Delta t)$
$\mathbf{p}_2(t)$
$\mathbf{p}_2(t+\Delta t)$
$\mathbf{p}_1(t)$
$\mathbf{p}_1(t+\Delta t)$
ω
Center point

Fig. 3: Pure Yaw Rotation

$$\bar{\mathbf{d}} = \mathbf{\Lambda}_p^{-\frac{1}{2}}\mathbf{U}_p^T\mathbf{d},$$

which satisfies

$$\bar{\mathbf{d}} \sim \mathcal{N}(\mu_{\bar{d}}, \mathbf{I}_2).$$

where

$$\mu_{\bar{d}} = \mathbf{\Lambda}_p^{-\frac{1}{2}}\mathbf{U}_p^T(\mu_p-\mathbf{q}(t+\Delta t)). \qquad (11)$$

Intuitively, the smaller $\mu_{\bar{d}}$ is, a higher probability the pairs $(\mathbf{p}(t), \mathbf{q}(t+\Delta t))$ matches with. Mathematically, the matching probability of the pairs $(\mathbf{p}, \mathbf{q}(t+\Delta t))$ is given as follows. Given α, $P(\|\mathbf{d}\| < \alpha)$ represents the matching probability. If

$$P(\|\bar{\mathbf{d}}\| < \alpha) \geq \beta,$$

where β is a given value, then the matching pair is accepted. Otherwise, the matching pair is eliminated. Since $\bar{\mathbf{d}} \sim \mathcal{N}(\mu_{\bar{d}}, \mathbf{I}_2)$, one has

$$P(\|\bar{\mathbf{d}}\| < \alpha) = \int_{\|\bar{\mathbf{d}}\|<\alpha} \frac{1}{2\pi} e^{-\frac{1}{2}(\mathbf{x}-\mu_{\bar{d}})^T(\mathbf{x}-\mu_{\bar{d}})} \mathbf{dx}.$$

Therefor, if $\|\mu_{\bar{d}}\| \leq h(\alpha,\beta)$, then $P(\|\bar{\mathbf{d}}\| < \alpha) \geq \beta$ holds, where $h(\alpha,\beta)$ is a function with respect to α, β and $h(\alpha,\beta) \geq 0$. Take a 1-dimension distribution for example: the probability's calculation can transform to integral of a standard gaussian distribution as shown in Fig.4. The probability is equal to integral of the curve from $-(\mu_{\bar{d}}+\alpha)$ to $-(\mu_{\bar{d}}-\alpha)$. As shown, when the $\mu_{\bar{d}} = 0$, the probability is equal to the area surrounded by red lines, and when $\mu_{\bar{d}} > 0$, the probability is equal to the area surrounded by black lines. It is obvious that the smaller $\mu_{\bar{d}}$ is, the larger the probability is. If it is larger than β, then the matching pair is accepted. So, it is reasonable that if $\|\mu_{\bar{d}}\| \leq h(\alpha,\beta)$, then $P(\|\bar{\mathbf{d}}\| < \alpha) \geq \beta$ holds. However, the computation $h(\cdot)$ is complicated. In practice, a simple way can be used. Given a parameter $\gamma \geq 0$, if

$$\|\mu_{\bar{d}}\| \leq \gamma$$

then the matching pair is accepted. Otherwise, the matching pair is eliminated.

4.4 Procedure to Error Matching Eliminating

```
algorithm of the method
Input:  all initial matching pairs, $\mathbf{v}^m, \mathbf{w}^m, {}^c\mathbf{R}_w, h$
Output: remained matching pairs after eliminating
for all the point pairs $(\mathbf{p}_i[u_{1i}, v_{1i}], \mathbf{q}_i[u_{2i}, v_{2i}])$ do
    $\mathbf{p}_i[u_{1i}, v_{1i}] \rightarrow \mathbf{p}[u_1, v_1]$
    $\mathbf{p}_i[u_{2i}, v_{2i}] \rightarrow \mathbf{p}[u_2, v_2]$
    substitute $[u_1, v_1]$ and $[u_2, v_2]$ to (3)
    respectively to get $[x_1, y_1]$ and $[x_2, y_2]$.
    obtain $Z$ through substitute $[x_1, y_1]$ and ${}^c\mathbf{R}_w$
    into the equation (7)
    get $\dot{\mathbf{p}}$ through substituting $Z, \mathbf{v}, \mathbf{w}$ and $[x_1, y_1]$
    to equation (6)
    get $x_{1new}$ and $y_{1new}$ by (9)
    get $\mathbf{\Sigma}_{p(t+\Delta t)}$ by (10)
    obtain $\mathbf{\Lambda}_p$ and $\mathbf{U}_p$ by extracting the eigenvalue
    of $\mathbf{\Sigma}_{p(t+\Delta t)}$
    obtain $\mu_{\bar{d}}$ by equation (11)
    find a appropriate value $\gamma$
    if $\|\mu_{\bar{d}}\| \leq \gamma$
        reserve the points pair
    else
        eliminate the points pair
    end if
end for
```

5 Experiments

The proposed method is tested on synthetic data of both simulation points and real images. First, performance of the proposed method is evaluated by simulation points and then compared with a RANSAC algorithm provided in the matlab toolbox [12] in Section 5.1. Then, in Section 5.2, a real experiment is carried out to demonstrate the effective of the proposed method.

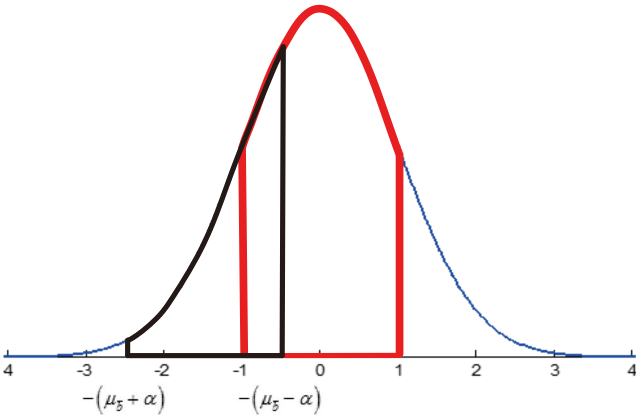


Fig. 4: 1-Dimension Distribution Model

5.1 Test on Simulation Points

The simulation is implemented by a virtual camera, whose image size is 640 pixels × 480 pixels. The intrinsic parameters of the camera are

$$\begin{aligned}\alpha_x &= 260, \quad \alpha_y = 260\\ u_0 &= 376, \quad u_0 = 240.\end{aligned}$$

Simulation points are randomly generated in the area $[-400, 400] \times [-400, 400] \times \{0\}$, and the distance between the points plane and the camera is 10 meter. The sampling time is $\Delta t = 0.05$s.

Image points of the first picture are obtained with a given setting camera pose and position. Then the virtual camera moves with a given speed $\mathbf{v}$ and an angular speed $\mathbf{w}$ during the time interval $[t, t + \Delta t]$, and takes the second picture. In the two pictures, N point pairs are obtained. In order to test the performance of the proposed method and the RANSAC algorithm, the latter half part of these point pairs is disorganized completely.

Table I. Simulation Result

	1	2	3
$\mathbf{v}$/(m/s)	$[3,3,3]^T$	$[5,5,5]^T$	$[8,8,8]^T$
$\mathbf{w}$/(°/s)	$[5,5,5]^T$	$[10,10,10]^T$	$[20,20,20]^T$
$\sigma_\mathbf{v}$	1	1	1
$\sigma_\mathbf{w}$	1	1	1
N	516	496	478
Proposed Method			
r_1	100%	100%	100%
r_2	1.5%	2.02%	0%
run time	0.0589s	0.0558s	0.0541s
RANSAC			
r_1	100%	81.05%	5.86%
r_2	100%	0.4%	0%
run time	0.2469s	0.2328s	0.2329s

In order to evaluate the performance, two rates denoted by r_1 and r_2 are chosen, defined as

$$r_1 = 2\frac{N_1}{N}, r_2 = 2\frac{N_2}{N},$$

where N_1 and N_2 are the number of reserved matched point pairs in the first half part and the latter half part, respectively. In theory, a good method will lead to r_1 closed to 1, and r_2 closed to 0, vice versa. Performances of the proposed method and the RANSAC algorithm are shown in the Table I. As shown, the proposed method has a better accuracy over the RANSAC method in most cases. More importantly, the proposed method can save much time compared with the RANSAC algorithm.

5.2 Test on Real Images

The experiment is carried out with a series of consecutive images. A virtual camera, same as in the simulation in Section 5.1, is mounted on a virtual quadrotor provided by a matlab toolbox [13]. Image points of the first picture are obtained with a given setting camera pose and position. Then the quadrotor together with the virtual camera moves at a given speed $\mathbf{v}$ and an angular speed $\mathbf{w}$ during the time interval $[t, t + \Delta t]$, and takes the second picture. With respect to the body frame of the quadrotor, the camera's installation parameter is

$${}^c\mathbf{R}_b = \mathbf{I}_3, {}^c\mathbf{T}_b = \mathbf{0}_{3\times 1}$$

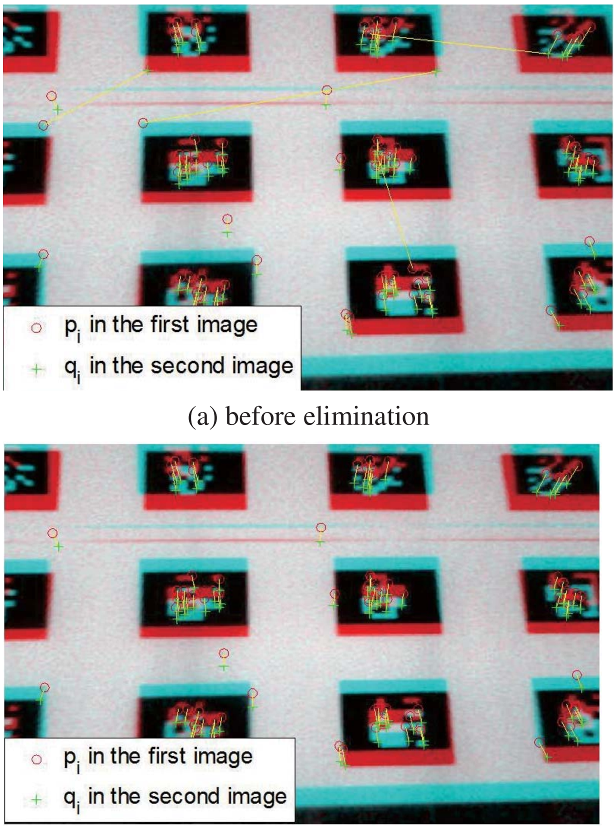
and the sampling period is $\Delta t = 0.05s$.

The first step of the experiment is to find the initial matching points $(\mathbf{p}_i, \mathbf{q}_i)$ between the two images. First, find the corner points by using the Harris-Stephens algorithm in the two images. Then derive the feature vectors of the corner points, and find the initial matching points whose features most likely to correspond between the two feature sets. The second step is to compute $\mathbf{v}$ and $\mathbf{w}$ through the obtained data with the aid of the toolbox. Then compute the matching probability density distribution of every $\mathbf{p}_i$ to judge whether $\mathbf{q}_i$ is a possible matching point with $\mathbf{p}_i$ or not. If it is possible, then the matching pair will be reserved. Otherwise, it will be eliminated.

Fig. 5 shows the matching results before and after the elimination, respectively. Each one of them consists of two stacked images. The circle points represent $\mathbf{p}_i$ in the first image, and the cross points represent $\mathbf{q}_i$ in the second image. Corresponding points in these two images are linked by a yellow line. If the error matching pairs in Fig. 5(a) cannot be shown in Fig. 5(b), then the effectiveness of method is shown. A conclusion can be observed from the experiment results: the proposed method can eliminate the error matching pairs obviously. This demonstrates the effectiveness of the proposed method.

6 Conclusion

This paper proposes a new method to eliminate mismatching points of every two successive images based on motion information. It is applicable to the cases that interest points are all from a plane. The superiority of the proposed method over RANSAC algorithms, the most commonly used error matching eliminating methods, is that it is non-iterative and thus time-efficient. The simulation illustrate the advantage of the proposed method over the RANSAC algorithm: it has a better accuracy and can save much time compared with the RANSAC algorithm. Then the real image experiment shows the effectiveness of the proposed method: it can eliminate the error matching pairs efficiently.


○ p_i in the first image
+ q_i in the second image
○ p_i in the first image
+ q_i in the second image

(a) before elimination

(b) after elimination

Fig. 5: Experiment Result

References

[1] S. Lee, T. Lee and S. Park, Flight test results of UAV automatic control using a single-antenna GPS receiver, in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, 2003: 33-36.

[2] H. Chao, M. Baumann, A. Jensen, Band-reconfigurable multi-UAV-based cooperative remote sensing for real-time water management and distributed irrigation control, in *IFAC World Congress*, 2008: 11744-11749.

[3] J. Wendel, An integrated GPS/MEMS-IMU navigation system for an autonomous helicopter, *Aerospace Science and Technology*, 10(6): 527-533, 2006.

[4] A. Cesetti, A vision-based guidance system for UAV navigation and safe landing using natural landmarks, in *2nd International Symposium on UAVs*, 2010: 233-257.

[5] G. Conte, P. Doherty, An integrated UAV navigation system based on aerial image matching, in *IEEE Aerospace Conference*, 2008: 1-10.

[6] M. A. Fischler, R. C. Bolles, Random Sample Consensus: a paradigm for model fitting with applications to image analysis and automated cartography, *Communications of the ACM*, 24(6): 381-395, 1981.

[7] W. Feller, *An Introduction to Probability Theory and Its Applications*. Hoboken: John Wiley & Sons, 2008.

[8] R. Hartley, A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge: Cambridge University Press, 2003.

[9] P. J. Bristeau, F. Callou, D. Vissiere, The navigation and control technology inside the ar. drone micro UAV, in *18th IFAC World Congress*, 2011: 1477-1484.

[10] R. Mahony, V. Kumar, P. Corke, Multirotor aerial vehicles: modeling, estimation, and control of quadrotor, *IEEE Robotics & Automation Magazine*, 19(3): 20-32, 2012.

[11] D. Abeywardena, S. Kodagoda, G. Dissanayake, R. Munasinghe, Improved state estimation in quadrotor MAVs: a novel drift-free velocity estimator, *IEEE Robotics & Automation Magazine*, 20(4): 32–39, 2013.

[12] http://www.vlfeat.org/download.html

[13] P. Corke, *Robotics, Vision & Control: Fundamental Algorithms in Matlab*. Heidelberg, Berlin: Springer-Verlag, 2011.