

Entropy Flow-Aided Navigation

He Deng, Chao Pan, Tongxin Wen, Jianguo Liu

*(National Key Laboratory of Science and Technology, Huazhong University,
Wuhan, PRC)*

(Email: jgliu@ieee.org)

Integrating the visual navigation mechanism of flying insects with a nonlinear Kalman filter, this paper proposes a novel navigation algorithm. New concepts of entropic map and entropy flow are presented, which can characterize topographic features and measure changes of the image respectively. Meanwhile, an auto-selecting algorithm of assessment threshold is proposed to improve computational accuracy and efficiency of global motion estimation. The simulation results suggest that the navigation algorithm can perform real-time rectification of the missile's trajectory well, and can reduce the cost of the missile's hardware.

KEY WORDS

1. entropy flow.
2. optical flow.
3. navigation.

1. INTRODUCTION. There are some flaws in the existing navigation methods. An inertial navigation system (INS) is a fully autonomous navigation system, but it is inappropriate for long-distance navigation [1]. In order to improve the accuracy of INS, some assistant navigation techniques including Doppler-aided, stellar-aided, radio-aided and terrain-aided techniques have been developed over the years though they do not generally perform well for low-velocity and highly manoeuvrable aircraft such as helicopters [2]. At present, GPS has been recognized as the ideal airborne navigation system, however, it also has some defects [3]. Therefore, developing other new navigation techniques is the focus of the research on future navigation technology. Recently, scholars have begun to explore new navigation methods learned from the visual navigation mechanisms of biology since certain insects, despite possessing relatively small brains and simple nervous systems, provide a clear demonstration that the living organism can display surprisingly competent mechanisms for guidance and navigation. Their quick, accurate and reliable visual navigation systems have an amazing capability to navigate accurately in complex natural environments, going without the influence of weather, climate, light or other environmental conditions. They are much better than any other current developments of human navigation systems [4,5]. If the visual navigation mechanisms of insects are studied in depth, and then applied to the navigation system of aircraft, robots or other equipment, some new bionic machines may be developed which could avoid obstacles and reach the target accurately in complex situations, just like some insects.

Many scholars have researched the visual biological mechanisms and proposed a number of corresponding navigation methods. Collett [6] has studied the behaviour of bees and ants, and proposed a landmark-matching model for robot navigation. Wehner [7] has mainly studied the navigation mode of desert ants and developed a wireless robot of the bionic desert ant. Srinivasan [8] has studied the visual navigation mechanism of flying insects, especially on bees for more than 20 years, and proposed a bionic calculation model of optical flow. Carroll [9] has studied the imaging mechanism of honeybees' compound eyes in depth and developed a bionic heuristic camera and a bionic motion detector.

Flying insects such as honeybees can perceive various patterns of movements that are generated by their own motion. And research suggests that insects rely on 'visual flow' to avoid lateral obstacles [10], to control their speed [8] and height [11,12], to cruise and land [8,12]. In this paper, in order to measure the changes of 'visual flow' accurately, we will propose new concepts of entropic map and entropy flow, which characterize topographic features and measure changes of images respectively. A six-parameter affine motion model to estimate motion parameters of the camera is also presented, according to the above two concepts. Meanwhile, a novel auto-selecting algorithm of assessment threshold is proposed to improve computational accuracy and efficiency of global motion estimation. Based on the above fundamental work, we will establish a new navigation algorithm based on the entropy flow and a Kalman filter. Compared with the traditional terrain-aided navigation, the proposed algorithm requires neither INS to provide location information nor an altimeter to provide height information. It will provide a feasible study for the navigation system of a missile.

The remainder of the paper is organized as follows: Two new concepts of entropic map and entropy flow are introduced in section 2, followed by the global motion estimation algorithm in section 3. The new navigation algorithm based on the entropy flow and Kalman filter is proposed in section 4. Some simulation experiment results are obtained in section 5 and finally the conclusions are given in section 6.

2. THE CONCEPTS OF ENTROPIC MAP AND ENTROPY FLOW.

Flying insects such as honeybees are well known to have high visual acuity and to use cues of image flow induced by their own motion to navigate, mate and identify food sources. The image flow can be explained as a kind of information flow since the image can be characterized by a set of information. As the information entropy is an effective means to depict image information, then the image flow can be interpreted as a kind of entropy flow.

2.1. *The visual mechanism of flying insects.* Honeybees are species of compound-eye insects. These compound eyes are composed of tens of thousands of ommatidia and the most obvious advantage of the ommatidium is that flying insects can effectively calculate the relative positions and distances from themselves to the observed objects, and then make their own judgments and response quickly. The observed range of each ommatidium is very narrow and the observed scenes overlap with each other between the adjacent ommatidia. The entire scene observed by all the ommatidia forms a panoramic image. Unlike vertebrates, flying insects cannot infer the distances to objects or surfaces due to their immobile eyes with fixed-focus optics. Thus they fall back on image motion, induced by their own motion, to deduce the distances to obstacles and to control various manoeuvres [7,10].

Research on the orientation performances of honeybees has agreed that honeybees determine the direction of the food source by using the sun compass, but there still exists controversy with respect to the possible cues used to measure the distance to the food source. Karl von Frich [13] proposed a theory of “energy hypothesis” that the honeybees infer the distance to the food source by measuring the amount of energy spent on the foraging trip. However, recent studies have discovered a number of different ways in which honeybees use visual cues for navigational purposes [7,10,11]. At the same time, some evidence has confirmed that the flight speeds of flying insects are regulated by monitoring the velocity of the image through their retinas [14,15]. Some findings [16] have proved that the distance is indeed estimated in terms of the total extent of image motion that is experienced by the eyes en route to the destination.

2.2. The concept of entropic map.

2.2.1. *Entropy.* A theoretical approach to information is based on the concept of entropy introduced by Shannon in information theory [17]. Suppose that a source X has L source alphabets and the probability of the i -th source x_i is given by p_i , where $p_k \geq 0$ ($k = 1, 2, \dots, L$) and $\sum p_k = 1$. An effective means to describe the information for the source X is the mean of self-information over the L source alphabets, which is called Shannon entropy.

$$H(X) = \sum_{i=1}^L p(x_i)I(x_i) = \sum_{i=1}^L p(x_i)[- \log(p_i)] = - \sum_{i=1}^L p(x_i)\log p_i \quad (1)$$

As an image is viewed as an information source with a probability vector described by its histogram L of the image I , the entropy of the histogram can be used to represent a certain level of information contained in the image. The entropy of an image is a scalar and it represents the total information content contained in the image.

2.2.2. *Local entropy.* Shannon entropy is a global quantity, considering only global statistical information while discarding the spatial distribution information of the image. Images with the same entropy may be completely different in vision. And in natural images, the difference of their entropies is very slight. Therefore, Shannon entropy is inappropriate to characterize the topographic features and measure changes of images. In order to overcome that deficiency, the concept of local entropy is proposed in [18]. For the sake of calculating local entropy, the image is partitioned into blocks of equal size. The local entropy considers the spatial distribution information of an image, and it denotes local information contained in local windows rather than global information contained in the whole image. So local entropy not only characterizes topographic features but also measures the changes of the size to some extent. All local entropies will construct a local entropic map, but its contrast and size are commonly different from the original image.

2.2.3. *The concept of entropic map.* The graph of function $y = -x \log x$ is shown in Figure 1. The function has a maximum point $1/e$ and it has two monotone intervals. These properties result that the contrast of the local entropic map is often out of accord with the original image. If the variable x is multiplied by $1/e$, the function will be limited to the interval $[0, 1/e]$ and increase monotonically in the interval $[0, 1/e]$, which is valid to avoid altering the contrast of the image. At the same time, the grey value $f(i, j)$ at the point (i, j) can be interpreted as the number of photons arriving at

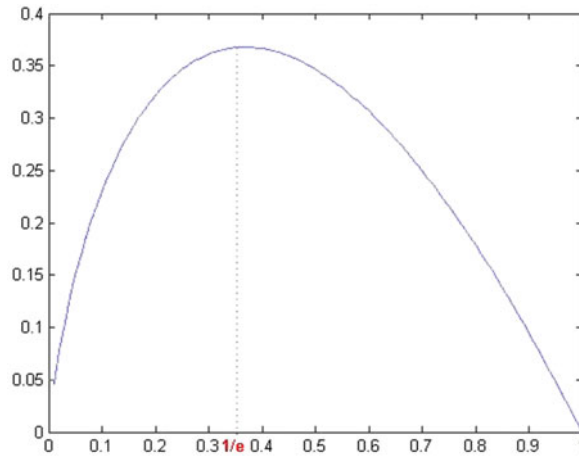


Figure 1. The graph of function $y = -x \log x$.

that point. If the total number of photons is viewed as 1, the grey value $f(i, j)$ will be interpreted as the probability distribution of photons, i.e. $p(i, j)$. According to these opinions, the choice of $p(i, j)$ in our entropic map is described as:

$$p(i, j) = \frac{f(i, j)}{e \cdot \sum_{i=1}^M \sum_{j=1}^N f(i, j)} \quad (2)$$

where $f(i, j)$ expresses the grey value at the point (i, j) and $M \times N$ means the size of the whole image.

In constructing the entropic map, the segmented method simulates the principle of compound eyes of flying insects. For an image of size $M \times N$, the constructing algorithm of entropic matrix T is described as follows:

- (1) Calculate $p(i, j)$ at each point (i, j) according to equation (2);
- (2) Partition the image into equal-sized rectangular sub-blocks (the number of blocks is $P \times Q$), and the adjacent sub-blocks will share a certain percentage of overlapping;
- (3) Compute entropy H_{ij} ($1 \leq i \leq P$, $1 \leq j \leq Q$) of each sub-block according to equation (1);
- (4) Use all sub-blocks of entropy to form an entropic matrix T .

The entropic matrix T is shown

$$T = \begin{bmatrix} H_{11} & H_{12} & \cdots & H_{1Q} \\ H_{21} & H_{22} & \cdots & H_{2Q} \\ \vdots & \vdots & \vdots & \vdots \\ H_{P1} & H_{P2} & \cdots & H_{PQ} \end{bmatrix} \quad (3)$$

If the entropic matrix T is normalized via the linear normalization method, the entropic map will be established. The entropic map can characterize topographic features of an image and its accuracy is relative to the fine grit of the sub-block.

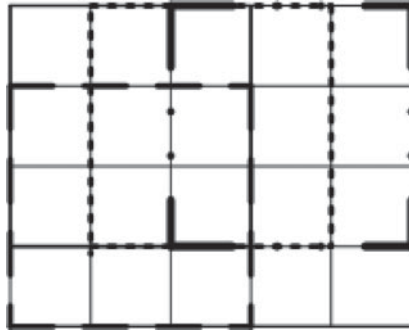


Figure 2. The diagram of intersected method.

In this paper, the original image is divided into blocks of size 3×3 and the overlapping ratio between two horizontal and vertical adjacent blocks is $2/3$. The diagram of the intersected method is shown in Figure 2.

Figure 3 (a) is an original airport image, and we have added Gaussian noise with zero mean and variance $var = 0.5$ to it, as shown in Figure 3 (b). Their corresponding entropic maps are shown in Figure 3 (c) and (d) respectively (via normalization). From Figure 3, we can see that entropic maps are insensitive to noise, similar to low-pass filter.

2.3. Entropy flow.

2.3.1. The concept of entropy flow. Recent research has discovered a number of different ways in which flying insects such as honeybees utilize self-induced image motion parameters for navigation purposes [4,8–11,19]. Optical flow is an approximation to image motion defined as the projection of velocities of 3-D surface points onto the imaging surface. The movement is reflected by the changes of brightness patterns of the image.

In information theory, an image can be viewed as an information source, i.e. a set of information. Then the image motion can be characterized by a kind of information motion. Entropy is a regular means to describe information. If the observed scene of each ommatidium is viewed as an element of the information set, i.e. the information of a small sub-block of the image, which is depicted by the entropy, the grey value of the image will be translated into an entropy model. Then, an entropic map can be constructed, imitating the principle of compound eyes. When a grey scale image is translated into an entropic map, the changes of brightness patterns will become the changes of entropy patterns. That is to say, the movement of the image is also characterized by the changes of entropy patterns called entropy flow. Not surprisingly, flying insects may use cues of entropy flow to manage the problems of navigation.

2.3.2. The computation of entropy flow. Optical flow is an approximate descriptor of image motion. Starting with the classical works of Horn and Schunck [20] as well as Lucas and Kanade [21], there are tremendous computational methods of optical flow. Barron et al. [22] suggested that the phase-based technique of Fleet and Jepson [23] and the differential technique of Lucas and Kanade [21] produced the more accurate results overall. Galvin et al. [24] concluded that the technique developed by Lucas and Kanade yields the best results.

In order to analyze motion within subsequent frames of an image sequence, temporal constancy has to be imposed on certain image features. The most frequently

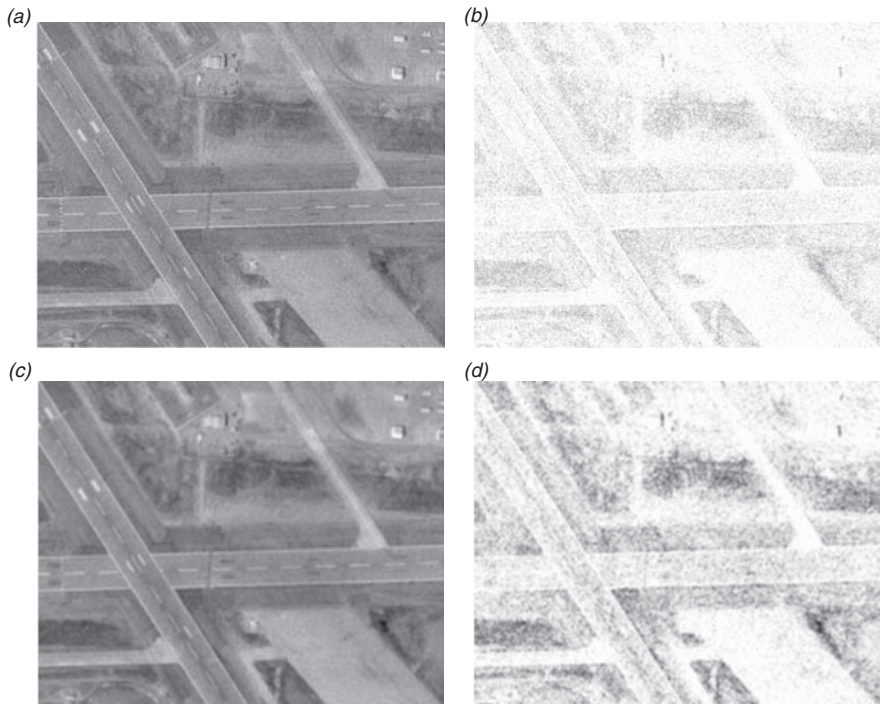


Figure 3. (a) *Top left*: Original airport image. (b) *Top right*: The image degraded by Gaussian noise with $var=0.5$. (c) *Bottom left*: Entropic map corresponds to the original image. (d) *Bottom right*: Entropic map corresponds to the noise image.

used feature in this context is the image brightness, i.e. the grey values of image objects in subsequent frames do not change over time. The brightness constancy assumption is also accepted to compute entropy flow in this paper, which makes the instantaneous entropy value in an entropic map sequence stay unchanged over time:

$$T(x+u, y+v, t+1) = T(x, y, t) \quad (4)$$

where the displacement field $\mathbf{v} = (u, v)^T(x, y, t)$ is entropy flow. For small displacements, we may perform a first order Taylor expansion to yield the entropy flow constraint (EFC):

$$T_x \cdot u + T_y \cdot v + T_t = 0 \quad (5)$$

where subscripts denote partial derivatives. It is evident that this single equation is not enough to uniquely compute the two unknowns u and v (aperture problem).

For the solution of equation (5), we add a local constant model to \mathbf{v} , similar to Lucas and Kanade [21]. Assume that in a small neighbourhood space Ω , the motion vectors remain constant, then the entropy flow can be estimated through weighted least-squares method. In a small spatial neighbourhood Ω , the estimation of entropy flow is computed by minimizing

$$\sum_{(x,y) \in \Omega} W(x,y) (T_x \cdot u + T_y \cdot v + T_t) \quad (6)$$

Table 1. The averages and standard deviations of angular errors for four synthetic image sequences.

	Av. Err.		St. Dev.	
	Optical flow	Entropy flow	Optical flow	Entropy flow
Sphere Sequence	8-1369° ± 0-1215°	8-1056° ± 0-1196°	15-1103° ± 0-1616°	14-8862° ± 0-1324°
Office Sequence	13-1347° ± 1-1661°	11-7536° ± 0-8271°	14-2982° ± 1-0789°	14-2800° ± 1-0684°
Street Sequence	16-2558° ± 1-9829°	15-1405° ± 1-8529°	23-1403° ± 1-7132°	23-0376° ± 1-5628°
Yosemite Sequence	14-9189° ± 1-4517°	14-2207° ± 0-9686°	19-6631° ± 1-2507°	18-4050° ± 0-5471°

where $W(x,y)$ represents a window. Then the solution of u and v is as follows

$$v = (A^T W^2 A)^{-1} A^T W^2 B \tag{7}$$

where in the time t , $X_i \in \Omega$ and

$$\left. \begin{aligned} A &= [\nabla T(X_1) \quad \nabla T(X_2) \quad \dots \quad \nabla T(X_n)] \\ W &= \text{diag}[W(X_1) \quad W(X_2) \quad \dots \quad W(X_n)] \\ B &= -[T_t(X_1) \quad T_t(X_2) \quad \dots \quad T_t(X_n)] \end{aligned} \right\} \tag{8}$$

2.3.3. *The performance evaluation of entropy flow.* Entropy flow is also an approximation of image motion. We compare entropy flow with optical flow in the estimated performance of image motion for four synthetic image sequences, since their 2-D image motion fields are known. In computation of optical flow, temporal smoothing is usually needed to avoid aliasing, but numerical differentiation must be done carefully [25]. In this paper, a Gaussian smoothing kernel with standard deviation $\sigma=1$ is used in the computation of optical flow, while that process is not necessary in calculation of entropy flow since the entropic map is robust to noise from section 2.2.

Three synthetic sequences, i.e. sphere sequence, office sequence and street sequence, are available from www.cs.otago.ac.nz/research/vision/ while the fourth sequence, Yosemite without clouds sequence is available from <http://www.cs.brown.edu/people/black/images.html>. The qualitative performance of image motion field is authenticated by the quantitative evaluations, i.e. measuring in terms of the error measure [23], namely the angular error between the correct image motion (u_c, v_c) and the estimated motion (u_e, v_e) via

$$\varphi_e = \arccos \left[\frac{u_c u_e + v_c v_e + 1}{\sqrt{u_c^2 + v_c^2 + 1} \sqrt{u_e^2 + v_e^2 + 1}} \right] \tag{9}$$

where φ_e is the angular error. The comparisons between entropy flow and optical flow was by means of average angular error (Av. Err.) and standard deviation of angular error (St. Dev.). The average angular errors and their standard deviations of four synthetic image sequences are shown in Table 1. It suggests that the estimation performance of entropy flow as a whole surpasses that of optical flow from Table 1. The reason may be that entropy flow possesses a bionic property since it is based on the visual mechanism of the flying insects, so it yields more precise results.

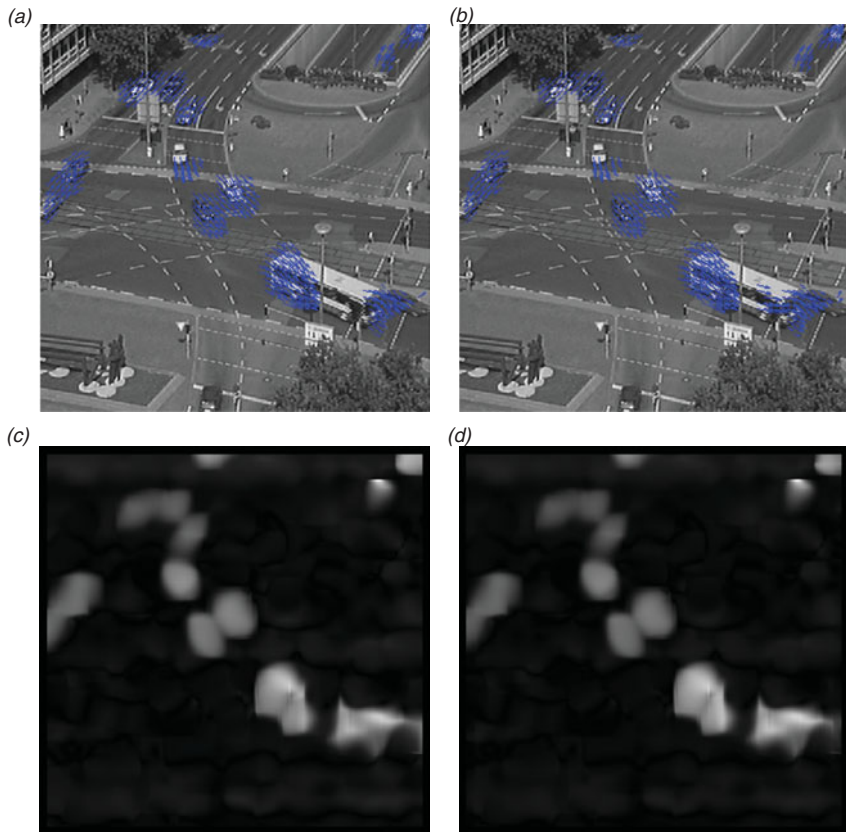


Figure 4. (a) *Top left*: Computed entropy flow field between frame 24 and 25. (b) *Top right*: Computed optical flow field. (c) *Bottom left*: The magnitude of entropy flow field. (d) *Bottom right*: The magnitude of optical flow field.

The comparisons for realistic image sequence Ettliger Tor Traffic sequence are shown in Figure 4. This sequence is available from http://i21www.ira.uka.de/image_sequences/, and it consists of 50 frames of size 512×512 . The computed entropy flow and optical flow field, as well as their corresponding magnitudes of flow field are shown in Figure 4. It suggests that entropy flow is very approximate to image motion, and perhaps it is better in depicting performance of image motion.

3. GLOBAL MOTION ESTIMATION ALGORITHM. Global motion is defined as the image motion induced by a camera moving through a static environment. The global motion estimation technique is simply described as follows: First choose a motion parameter model and then estimate those parameters through a robust parameter estimation method, e.g. least squares, M-estimators and Levenberg-Marquardt iteration algorithm. We can use entropy flow to estimate those parameters since entropy flow is an approximation to image motion.

3.1. Six-parameter affine motion model. The field generated by the 3-D rigid object motion is projected onto a 2-D motion field under the orthogonal projection

which can be depicted by both affine motion model and non-linear perspective motion model. With regard to global motion models, we adopt a six-parameter affine motion model because of its reasonable trade-off between complexity and accuracy. The basic form of six-parameter affine motion model is described as:

$$\begin{cases} u(x, y) = a_1x + a_2y + a_3 \\ v(x, y) = a_4x + a_5y + a_6 \end{cases} \tag{10}$$

where $\mathbf{v} = (u(x, y), v(x, y))$ represents the motion at the point (x, y) . If we assume $\alpha_u^T = (a_1, a_2, a_3)$, $\alpha_v^T = (a_4, a_5, a_6)$, $X = [x, y, 1]$, we will get $u(x, y) = X \cdot \alpha_u$ and $v(x, y) = X \cdot \alpha_v$. Obviously, when the motion \mathbf{v} is known, the parameters α_u and α_v will be estimated through a least-squares estimation algorithm because of its small computational complexity and good estimated accuracy. Then the solution of $a_1, a_2, a_3, a_4, a_5, a_6$ is as follows:

$$[\alpha_u \ \alpha_v] = \left[\sum_{(x,y) \in D} X^T \cdot X \right]^{-1} \sum_{(x,y) \in D} X^T \cdot [u(x, y) \ v(x, y)] \tag{11}$$

where D is a set of points in the entropic map.

3.2. *Auto-selecting algorithm of assessment threshold.* However, the estimated uncertainty and the violation of EFC have a great impact on the estimated accuracy of those motion parameters. In order to improve computational accuracy and efficiency of global motion estimation, an auto-selecting algorithm of assessment threshold is proposed in this paper. From equation (10), three motion vectors \mathbf{v} are enough to solve those motion parameters. But this strategy is not adopted because of large errors. The method that all motion vectors \mathbf{v} contribute to compute the motion parameters is also inadvisable owing to the volatility of entropy flow and bulky calculated amount. The aim of the auto-selecting algorithm of assessment threshold is to seek a small quantity of entropy flow with little volatility. Therefore, the entropy flow field is divided into sub-blocks of equal size, and the motion parameters are estimated in each sub-block. The estimated parameters in each sub-block are shown as:

$$[\alpha_{ui} \ \alpha_{vi}] = \left[\sum_{(x,y) \in D_i} X^T \cdot X \right]^{-1} \sum_{(x,y) \in D_i} X^T \cdot [u_i(x, y) \ v_i(x, y)] \quad i = 1, 2, \dots, M \tag{12}$$

where α_{ui}^T and α_{vi}^T denote the estimated parameters in i -th sub-block, D_i is a set of points in i -th sub-block, and M is the number of sub-blocks.

In order to evaluate the estimated performances of these sub-blocks, the estimated error is introduced in this paper. The estimated error of each sub-block is defined as:

$$\sigma_i^2 = \frac{1}{N_i} \sum_{(x,y) \in D_i} \|\mathbf{v}_i(x, y) - \mathbf{v}_{ai}(x, y)\|^2, \quad i = 1, 2, \dots, M \tag{13}$$

where N_i is the number of points in i -th sub-block, \mathbf{v}_i is the original entropy flow in i -th sub-block, \mathbf{v}_{ai} is the estimated entropy flow through estimated motion parameters, and M is the number of sub-blocks. Then, some sub-blocks whose estimated errors are small via a threshold are chosen to re-estimate the motion parameters, which may result in more accurate motion parameters and smaller calculated amount.

The selection of threshold is very important. In our simulation experiments, we found that the proportion of block selection has a great effect on the threshold. The

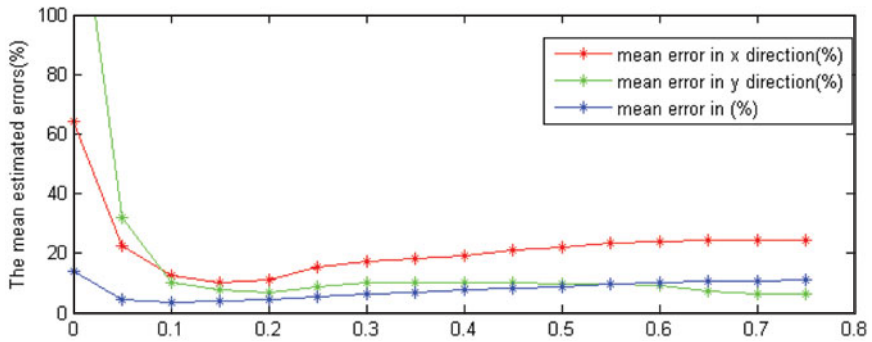


Figure 5. The respective mean errors under different proportions of block selection.

auto-selecting algorithm of assessment threshold is based on the proportion of block selection, and described in detail as follows:

- (1) Partition the entropy flow field into sub-blocks of equal size (the size usually is 8×8 or 16×16);
- (2) Calculate estimated errors of each sub-block, σ_i^2 , $i = 1, 2, \dots, M$, and M is the number of sub-blocks;
- (3) Set up an initial threshold $\Omega = \Omega_0$ and the step length $\Delta\Omega$;
- (4) Choose the sub-blocks whose estimated error satisfy $\sigma_i^2 \leq \Omega$, $i = 1, 2, \dots, N$, where N means the number of selected sub-blocks;
- (5) Put up the proportion of block selection β , if $N/M \geq \beta$, then Ω is the final threshold. Otherwise, $\Omega = \Omega + \Delta\Omega$, return to Step 2 and continue to search.

The respective mean of estimated errors of the translation in X , Y and Z coordinates are shown in Figure 5 (the block size is 16×16). The horizontal axis represents different proportion of block selection β and the vertical axis denotes the mean estimation error (the errors are described in percent). The total number of experiments is 288 and β is finally chosen as 0.15 in the following work.

4. THE NAVIGATION ALGORITHM BASED ON ENTROPY FLOW AND KALMAN FILTER. If a camera is fixed on the body of the missile, the motion parameters of the camera will be the same as those of the missile. It is possible to estimate motion state of the missile from section 3, so the control system of the missile can deal with some problems of navigation.

4.1. Kalman filter. The least-squares estimation algorithm is adopted to estimate the parameters of global motion. However, the least-squares algorithm is usually more sensitive to data noise. It is a linear minimum deviation estimation and its process needs to know the first-order and the second-order moments of the estimator exactly, which is very harsh, usually impossible for a non-stationary process. Therefore, the estimated results have a certain extent of error and need to be amended.

In the 1960s, the Kalman filter method [26] was proposed, which aroused worldwide application. The advantage is that it has transitivity, so it does not store large amounts of historical information and can reduce the amount of storage in the computer, directly combining the system state equation with observation equation,

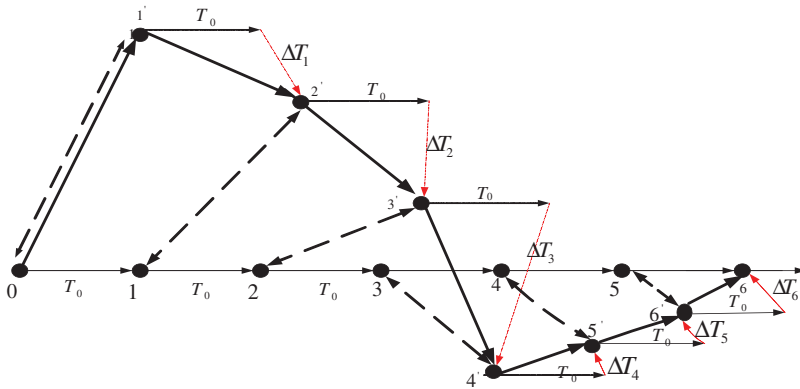


Figure 6. The sketch map of the navigation algorithm.

and estimating parameters of the system state while directly giving estimated accuracy. So it is widely used in process control, telecommunications and biomedical fields, etc.

4.2. *The framework of the navigation algorithm.* The novel navigation algorithm is modelled on the basis of the navigation mechanism of particular flying insects – honeybees. From section 2.1, the flight direction of honeybees to the food source is definite since they determine the direction by using the sun compass, which rouses us to plan the flight path of the missile beforehand (called reference trajectory). Meanwhile, honeybees use cues of image flow induced by their own motion to avoid lateral obstacles, to control their speed and height, to cruise and land, which inspires us to fix a high-performance camera on the abdomen of the missile. The aim of the camera is to snap the real-time image sequence and then expect to regulate the flight trajectory of the missile in real time (called real-time trajectory). Furthermore, the motion parameters of the camera, i.e. the real-time flight regime of the missile, are estimated through entropy flow, and then a series of aerial photographs from an original point (location) to the target point (location) are stored in the missiles beforehand (called reference image sequence). Under the above initial assumptions, the control system of the missile may continuously modulate its control signal to regulate the flight trajectory of the missile through the navigation algorithm based on the entropy flow and a Kalman filter, to reach the goal.

The computation of entropy flow is between real-time images and their corresponding reference images, rather than only in the reference image sequence or the real-time image sequence. The initial image in the reference image sequence is viewed as the first-frame image, while the initial image in the real-time image sequence is regarded as the corresponding second-frame image. Then the entropy flow filed will be calculated according to section 2, and the motion parameters of the missile are obtained based on section 3. The control system of the missile regulates the trajectory, and instructs the missile to reach the next target point (location). By repeating that process, the missile will arrive at the end-point. The navigation algorithm is a continuously iterative process to ensure arrival at the end-point. The sketch map of the navigation algorithm is shown in Figure 6.

In Figure 6, 0, 1, 2, ..., 6, ... represent the respective positions of the reference trajectory and they move in a unit time to reach the next position according to the scheduled speed and direction (i.e. T_0). The images in the position 0, 1, 2, ..., 6, ... are

viewed as the first frame, the second frame, ..., the sixth frame, ... in the reference trajectory, i.e. reference images. These images are converted into the entropic maps, which are denoted as $\mathbf{0}$, $\mathbf{1}$, $\mathbf{2}$, ..., $\mathbf{6}$, ... respectively. $\mathbf{1}'$, $\mathbf{2}'$, ..., $\mathbf{6}'$, ... represent the positions of the real-time trajectory and they move in a unit time to arrive at the next location according to the amendatory speed and direction (i.e. $T_0 + \Delta T_1$, $T_0 + \Delta T_2$, ..., $T_0 + \Delta T_6$, ...). The following work is similar to the above, denoted as $\mathbf{1}'$, $\mathbf{2}'$, ..., $\mathbf{6}'$, ... respectively.

The navigation algorithm based on the entropy flow and Kalman filter is described in detail as follows:

- (1) Compute entropy flow between $\mathbf{1}'$ and $\mathbf{0}$ according to section 2;
- (2) Estimate the motion parameters of the missile in the light of section 3;
- (3) Input the estimated parameters to Kalman filter to obtain the optimized motion parameters ΔT_1 ;
- (4) The position of $\mathbf{1}'$ moves to the position of $\mathbf{2}'$ through $T_0 + \Delta T_1$.

The translation in X and Y coordinates can be viewed as the velocity in X and Y coordinates since we only consider the displacements in X , Y and Z coordinates in a unit time interval in this paper. Return to step 1, and compute entropy flow between $\mathbf{2}'$ and $\mathbf{1}$, the optimized parameters ΔT_2 will be obtained, thus the position of $\mathbf{3}'$ is available by $\mathbf{2}'$ moving $T_0 + \Delta T_2$. Repeat the above steps, therefore a series of rectification ΔT_3 , ΔT_4 , ... will be obtained respectively until arriving at the goal.

5. EXPERIMENTS. In the simulation experiments, we assume that: 1) the camera is fixed on the missile and its optical axis is perpendicular to the ground; 2) the translation is only considered in X , Y and Z coordinates without considering rotary issues and the changes of the missile's attitude; 3) the body of the missile is always parallel to the ground and there is no roll-over phenomenon.

5.1. *Simulation process.* The reference images were snapped in five different initial locations at seven different heights, and the central positions of their images were controlled by the latitude and longitude in Google Earth, i.e. ($30^\circ 30' 35.03''$, $114^\circ 24' 29.04''$), ($23^\circ 09' 36.36''$, $113^\circ 14' 58.45''$), ($30^\circ 30' 20.22''$, $114^\circ 24' 44.24''$), ($39^\circ 54' 18.16''$, $116^\circ 24' 27.74''$), ($40^\circ 39' 10.33''$, $73^\circ 57' 34.33''$). At each central location, seven images were snapped respectively, from the height of 1850m to 2150m, with 50m as an interval. The total number of snapped images was 35 and the size was 689×1024 . The snapped images in different view heights were used for simulating the changes of the height, and forming other images which are in the other height through an interpolation algorithm in simulation experiments.

During the practical flight, random error is inevitable. In order to simulate random error, in the fourth step of the navigation algorithm in section 4.2, we have added random errors in the interval $[-0.5, 0.5]$ (unit: pixel) on the translations in X and Y coordinates and random errors in the interval $[-5, 5]$ (unit: metre) on the translation in Z coordinate in each iteration step. The initial location of the real-time trajectory is obtained from the initial position of reference trajectory where an initial error is added.

The simulation experiments at each initial position accompanied by five different initial errors are repeated 30 times randomly, then the total number of simulation experiments is 750. Five experimental results randomly singled out from five different

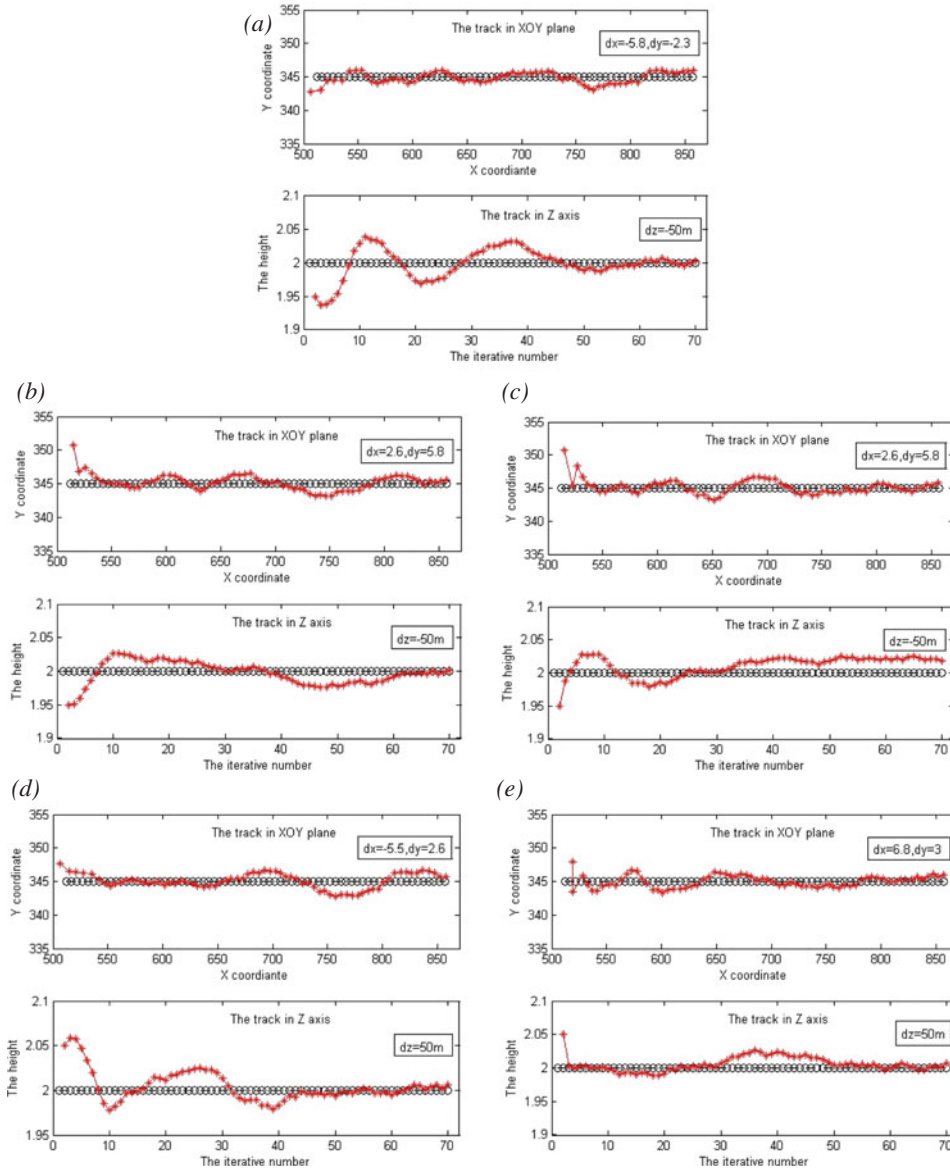


Figure 7. (a) *Top*: A result randomly taken out from (30°30'35.03", 114°24'29.04") and its initial error is (-5.8, -2.3, -50). (b) *Middle left*: From (23°09'36.36", 113°14'58.45") and the initial error is (2.6, 5.8, -50). (c) *Middle right*: From (30°30'20.22", 114°24'44.24") and the initial error is (2.6, 5.8, -50). (d) *Bottom left*: From (22°39'06.36", 113°29'58.45") and the initial error is (-5.5, 2.6, 50). (e) *Bottom right*: From (36°39'55.19", 116°59'41.69") and the initial error is (6.8, 3.0, 50).

initial positions are shown in Figure 7. Each image has two sub-images. The real-time amendatory trajectory in the XOY plane is shown in the first sub-image, where the red curve with volatility is the real-time trajectory and the grey line in the centre represents the reference trajectory. The real-time amendatory trajectory in the Z axis is shown in the second sub-image, where the red curve with volatility is the real-time

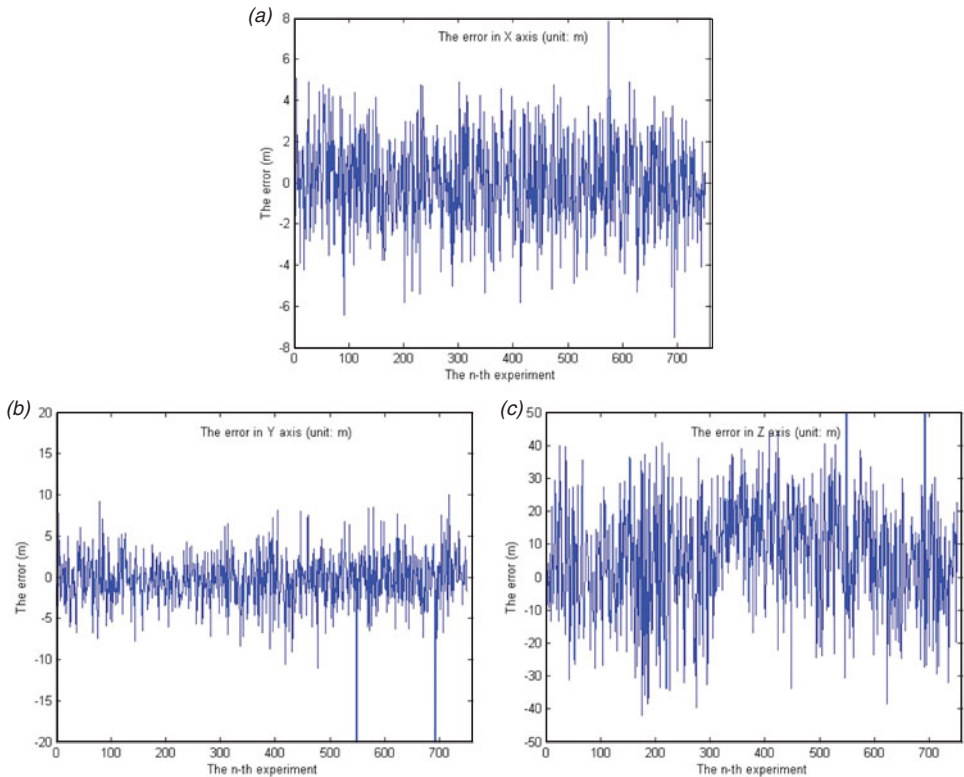


Figure 8. The respective errors in X , Y and Z coordinates from 750 times experiments.

modified height and the grey line in the centre is the reference height 2000m. The initial errors in X and Y coordinates were $(-5.8, -2.3)$, $(2.6, 5.8)$, $(2.6, 5.8)$, $(-5.5, 2.6)$, and $(6.8, 3.0)$ respectively (unit: pixel), while the respective initial errors in Z axis were 50m, -50 m, 50m, 50m and 50m (unit: metre).

5.2. *Circular error probability (CEP)*. In the military science of ballistics, circular error probable (CEP) or circular error probability is an intuitive measure of testing a weapon system's accuracy. In Pitman [27], CEP is defined "as the radius of a circle around a target for which there is a 50% probability that the guidance system will guide the vehicle into it." The following approximate method [27] has been developed to estimate the radius of probability distribution R since CEP is very difficult to evaluate. A plot of CEP/σ_y versus $\rho = \sigma_x/\sigma_y$ shows that the CEP/σ_y is fairly linear over the range $0.3 < \rho < 1.0$ and the CEP can be adequately represented in this region by the equation, $CEP = 0.615\sigma_x + 0.562\sigma_y$, where σ_y is the larger of the two errors. The relationship can be expressed as:

$$CEP = \begin{cases} 0.615\sigma_x + 0.562\sigma_y & (\sigma_x < \sigma_y) \\ 1.1774\sigma_x & (\sigma_x = \sigma_y) \\ 0.615\sigma_y + 0.562\sigma_x & (\sigma_x > \sigma_y) \end{cases} \quad (14)$$

The respective error distributions in X , Y and Z coordinates from 750 simulation experiments are shown in Figure 8, where the horizontal axis represents the n -th simulation experiment, and the vertical axis represents the error (unit: m). From

Table 2. The mean and variance of errors in the X-axis, Y-axis and Z-axis from 750 times experiments.

Initial position	Initial error	X-axis		Y-axis		Z-axis	
		mean (m)	variance (m)	mean (m)	variance (m)	mean (m)	variance (m)
30°30'35.03" N	(6, 1, -50)	-0.2886	2.3531	-1.0192	2.7701	4.7022	14.1628
114°24'29.04" E	(-5.6, 1.8, -50)	0.4646	2.4480	-0.2461	4.3556	5.7493	14.8642
	(-4.6, -2.5, 50)	1.1360	2.5777	-0.0045	3.1017	4.4728	15.1368
	(2.6, -5.5, 50)	0.8211	2.1572	-0.3898	3.5509	-1.190	17.2161
	(-1.7, 6.4, 30)	-0.2436	2.2232	-0.6506	2.6645	3.6932	15.4964
23°09'36.36" N	(6, 1, -50)	1.1406	2.2574	-1.4738	4.0485	8.6414	15.1939
113°14'58.45" E	(-5.6, 1.8, -50)	1.0942	2.2301	-1.2930	4.1547	7.8844	14.2527
	(-4.6, -2.5, 50)	0.3270	2.0937	-2.1914	3.7444	16.7295	16.6989
	(2.6, -5.5, 50)	3.9294	3.4501	-410.273	362.740	105.867	61.8940
	(-1.7, 6.4, 30)	0.8449	2.1348	-2.2524	4.0111	16.4336	18.8146
30°30'20.22" N	(6, 1, -50)	0.9724	2.2762	-1.6659	3.6004	9.3499	14.5423
114°24'44.24" E	(-5.6, 1.8, -50)	0.7389	2.5899	-0.6702	3.0956	10.7475	17.8651
	(-4.6, -2.5, 50)	-0.3264	2.1121	-2.3362	6.1217	24.6246	18.4045
	(2.6, -5.5, 50)	2.8213	3.8190	-267.953	355.501	76.6343	64.1384
	(-1.7, 6.4, 30)	0.3477	2.1685	-2.0441	4.1214	23.6664	17.5366
39°54'18.16" N	(6, 1, -50)	0.4975	2.2770	-1.6410	5.1098	17.7443	14.8620
116°24'27.74" E	(-5.6, 1.8, -50)	-0.2867	1.8134	-1.4610	4.0397	13.0358	13.4231
	(-4.6, -2.5, 50)	0.3284	2.6211	-1.2418	3.5002	17.8275	17.7935
	(2.6, -5.5, 50)	3.7131	3.6268	-407.822	361.630	103.915	61.2574
	(-1.7, 6.4, 30)	0.5193	1.4762	-2.7872	4.4614	15.4556	18.7414
40°39'10.33" N	(6, 1, -50)	0.6031	1.8384	-2.0247	4.5647	14.1980	13.8775
73°57'34.33" E	(-5.6, 1.8, -50)	0.8849	2.0298	-3.4207	4.3209	12.0327	17.0805
	(-4.6, -2.5, 50)	-0.0705	2.8921	-1.5034	4.9440	19.4714	15.8593
	(2.6, -5.5, 50)	3.6926	3.9365	-359.730	362.465	92.9032	65.0534
	(-1.7, 6.4, 30)	0.6834	1.4262	-1.9255	4.1115	13.2422	17.7181
The total mean and variance (m)		0.9738	2.7580	-1.6069	4.0766	12.8747	17.1830
(The larger errors have been removed)							
CEP (m)				3.9872			

Figure 8, the majority of translational errors in X, Y and Z coordinates are in the interval [-3m, 3m], [-5m, 5m] and [-20m, 20m] respectively. The width of each pixel in the image at the height of 2000m represents the size of 2.2559m in actual ground width. The majority of translational errors in X and Y coordinates are within two pixels while they are under 1% in Z coordinate. CEP is 3.9872m according to equation (14). The same result can also be obtained from Table 2.

The mean and variance of errors in the X-axis, Y-axis and Z-axis from 750 times experiments are shown in Table 2. The error unit in X and Y coordinates is pixel and the error unit in Z coordinate is metre. There exist large error points (marked data) though most of the errors in X, Y and Z coordinates are small from Table 2. Those marked datas have been removed in calculating the total mean and variance of error.

CEP is an intuitive measure of testing the accuracy of the missile system. Different initial errors may result in different CEPs, and some initial errors may produce large CEP which do not satisfy the accuracy requirements of the navigation system. In

Table 3. The mean and variance of errors under different initial errors in X and Y coordinates.

The resultant errors			The initial errors in Y axis								
			1 ~ 5(pixels)			5 ~ 10(pixels)			10 ~ 15(pixels)		
			mean (m)	variance (m)	CEP (m)	mean (m)	variance (m)	CEP (m)	mean (m)	variance (m)	CEP (m)
The initial errors in X axis	1 ~ 5 (pixels)	X	0.0053	2.7644	3.6652	0.0456	2.8312	3.6956	0.0236	3.3154	4.0406
		Y	-0.8615	3.4966		-0.8689	3.4776		-0.6028	3.5616	
	5 ~ 10 (pixels)	X	0.1046	2.7888	3.6909	-0.3562	3.3191	4.1702	—	—	388.0947
		Y	-0.7080	3.5156		-0.5459	3.7881		—	—	
	10 ~ 15 (pixels)	X	—	—	232.5683	—	—	337.9097	—	—	—
		Y	—	—		—	—		—	—	

Note: “—” denotes the value can not be computed.

order to study the range of initial error, many experiments were done. There were nine groups of experiments and each group was repeated 750 times. The initial error in Z coordinate is always in the interval $[-50\text{m}, 50\text{m}]$. The initial errors in X , Y and Z coordinates are chosen randomly. The experimental results are shown in Table 3. From Table 3, we can see that if the initial errors in both X and Y coordinates are under 10 pixels, the CEP will be less than 4.2m, which can meet the accuracy requirement. If the initial error in X coordinate or in Y coordinate is larger than 10 pixels, the CEP will become very large, and it cannot be calculated.

6. SUMMARY AND CONCLUSIONS. In this paper, we have analyzed the visual navigation mechanism of flying insects in depth, and thereupon proposed a novel navigation algorithm for a missile. The navigation algorithm can perform real-time adjustments of the missile's trajectories in the 3D space well only if the initial errors in both X and Y coordinates are under 10 pixels from simulation experiments. The navigation algorithm does not need the accelerometers or gyroscopes and it has provided a feasible method for the new navigation research.

We have presented two novel and essential concepts of entropic map and entropy flow, which can depict topographic features and estimate the motion of the image respectively. The performance of entropy flow is superior to that of optical flow in comparison.

We have proposed a new global motion estimation algorithm based on the entropy flow. The auto-selecting algorithm of assessment threshold is proposed to improve computational accuracy and efficiency of global motion estimation.

We have certainly not reached the end of the road yet since the new navigation algorithm only considers the translations in X , Y and Z coordinates. The algorithm does not take into account the rotations in X , Y and Z coordinates and the changes of attitude of the missile airframe, which are the focus of further studies.

ACKNOWLEDGMENT

This project is supported in part by the National Nature Science Fund of China under Grants 60672060 and 61071136 and the Research Fund for the Doctoral Program of Higher Education of China.

REFERENCES

- [1] Farrell, J. L. (1976). *Integrated Aircraft Navigation*. New York: Academic.
- [2] Enns, R. and Morrell, D. (1995). Terrain-aided navigation using the viterbi algorithm. *Journal of Guidance, Control and Dynamics*, **18**, 1444–1449.
- [3] Bertoni, G., Borghi, D. and Zanzi, M. (1998). The GPS integrity problem. *Proceeding of the 14th IFAC Symposium on Automatic Control in Aerospace*, Seoul, Korea.
- [4] Esch, H. E. and Burns, J. E. (1996). Distance estimation by foraging honeybees. *Journal of Experimental Biology*, **199**, 155–162.
- [5] Reisenman, C., Haag, J. and Borst, A. (2003). Adaptation of response transients in fly motion vision. I: Experiments. *Vision Research*, **43**, 1291–1307.
- [6] Collett, T. S., Dillmann, E., Giger, A. and Wehner, R. (1992). Visual landmarks and route following in desert ants. *Journal of Comparative Physiology A*, **170**, 435–442.
- [7] Wehner, R. (2003). Desert ant navigation: how miniature brains solve complex tasks. *Journal of Comparative Physiology A*, **189**, 579–588.
- [8] Srinivasan, M. V., Zhang, S. W., Lehrer, M. and Collett, T. S. (1996). Honeybee navigation en route to the goal: visual flight control and odometer. *Journal of Experimental Biology*, **199**, 237–244.
- [9] O'Carroll, D. C., Bidwell, N. J., Laughlin, S. B. and Warrant, E. J. (1996). Insect motion detectors matched to visual ecology. *Nature*, **382**, 63–66.
- [10] Srinivasan, M. V., Lehrer, M., Kirchner, W. H. and Zhang, S. W. (1991). Range perception through apparent image speed in freely flying honeybees. *Visual Neuroscience*, **6**, 519–535.
- [11] Baird, E., Srinivasan, M. V., Zhang, S., Lamont, R. and Cowling, A. (2006). Visual control of flight speed and height in honeybee. *Lecture Notes in Artificial Intelligence*, **4095**, 40–51.
- [12] Franceschini, N., Ruffier, F. and Serres, J. (2007). A bio-inspired flying robot sheds light on insect piloting abilities. *Current Biology*, **17**, 329–335.
- [13] von Frisch, K. (1967). *The dance language and orientation of honeybees*. Cambridge: Belknap.
- [14] David, C. T. (1982). Compensation for height in the control of groundspeed by *Drosophila* in a new, "Barber's Pole" wind tunnel. *Journal of Comparative Physiology A*, **147**, 485–493.
- [15] Baird, E., Srinivasan, M. V., Zhang, S. W. and Cowling, A. (2005). Visual control of flight speed in honeybees. *Journal of Experimental Biology*, **208**, 3895–3905.
- [16] Esch, H. E., Zhang, S. W., Srinivasan M. V. and Tautz, J. (2001). Honeybee dances communicate distances measured by optic flow. *Nature*, **411**, 581–583.
- [17] Shannon, C. E. (1948). A mathematical theory of communication. *ACM SIGMOBILE Mobile Computing and Communications Review*, **5**, 3–55.
- [18] Shiozacki, A. (1986). Edge extraction using entropy operator. *Computer Vision, Graphics and Image Processing archive*, **36**, 1–9.
- [19] Si, A., Srinivasan, M. V. and Zhang, S. (2003). Honeybee navigation: properties of the visually driven 'odometer'. *Journal of Experimental Biology*, **206**, 1265–1273.
- [20] Horn, B. and Schunck, B. (1981). Determining optical flow. *Artificial Intelligence*, **17**, 185–203.
- [21] Lucas, B. and Kanade, T. (1981). An iterative image registration technique with an application to stereo vision. *Proceedings of the 7th International Joint Conference on Artificial Intelligence*, Canada.
- [22] Barron, J. L., Fleet, D. J. and Beauchemin, S. S. (1994). Performance of optical flow techniques. *International Journal of Computer Vision*, **12**, 43–77.
- [23] Fleet, D. J. and Jepson, A. D. (1990). Computation of component image velocity from local phase information. *International Journal of Computer Vision*, **5**, 77–104.
- [24] Galvin, B., McCane, B., Novins, K., Mason, D. and Mills, S. (1998). Recovering motion fields: an analysis of eight optical flow algorithms. *Proceedings of 1998 British Machine Vision Conference*, Southampton, England.
- [25] Beauchemin, S. S. and Barron, J. L. (1995). The computation of optical flow. *ACM Computing Surveys*, **27**, 433–467.
- [26] Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Transactions of the ASME – Journal of Basic Engineering*, **82**, 35–45.
- [27] Pitman, G. R. (1962). *Inertial Guidance*. New York: Academic.