Pattern formation for a fleet of AUVs based on optical sensor

Xiaomin Wang, Benoit Clement, Benoit Zerr, Helene Thomas, Zexiao Xie

Pattern formation for a fleet of AUVs based on optical sensor

Xiaomin WANG
College of engineering,
Ocean University of China
Lab-STICC UMR CNRS 6285
ENSTA Bretagne
Qingdao, China 266100
Email:xiaomin.wang@ensta-bretagne.org

Benoit ZERR
Lab-STICC UMR CNRS 6285
ENSTA Bretagne
Brest, France 29200
Email: benoit.zerr@ensta-bretagne.fr

Hélène THOMAS
ENSTA Bretagne
Brest, France 29200
Email: helene.thomas@ensta-bretagne.fr

Benoit CLEMENT
Lab-STICC UMR CNRS 6285
ENSTA Bretagne
Brest, France 29200
Email: benoit.clement@ensta-bretagne.fr

Zexiao XIE
College of engineering,
Ocean University of China
Qingdao, China 266100
Email: xiezexiao@ouc.edu.cn

***Abstract*—During the past decades, multi-robot systems have been studied deeply and have demonstrated their advantages in conducting autonomous missions, particularly in underwater environment. When using multiple robots, high cost single sensor can be replaced by many coordinated low cost sensors, such as optical sensors used in our work for acquiring relevant data and coordinating the fleet of robots. In this paper, we propose a coordination algorithm for pattern formation (the shape of the fleet), which includes a new predefined pyramid pattern independent on the number of robots and an associated formation control strategy for building and maintaining this pyramid without collision among robots and without external assistant in non-obstacle situation. The formation control strategy consists of three steps: (1) a common frame is built after the exchange of local information extracted from visual servoing (VS) and compass of each robot; (2) a pyramid frame is established from the distribution of robots, and whose orientation is defined by one of the two following methods: one based on principal component analysis (PCA) and the other one based on an intermediate circle pattern (CP); (3) the kernel part is the collision avoidance strategy (CA) which is realized by optimally matching up positions in the distribution and the pyramid, and planning the straight non-intercrossed trajectories. At last, robots build the pyramid along the straight trajectory. During the whole process, except the information exchange for establishing the common frame, robots do not need communication as they have the same algorithms. Until now the performance of the formation control strategy is demonstrated with 4 to 6 robots in Blender based on dynamic equations of real underwater robots.**

I. Introduction

Multi-robot systems have been widely researched and utilized because of its merits: high reliability, increasing extensibility, enhanced flexibility, satisfying efficiency, simplicity and redundancy compared with single sophisticate robot [1]. The extent of underwater environment is particularly suited for multi-robot missions (wreckage detection, oil pipe monitoring, ocean sampling and mapping, etc.). For example, multi-AUV system has more advantages than single AUV for the exploration of deep sea. Robots can help each other out of trouble, out of losting direction etc. Even if one robot is broken, others can continue to accomplish the task. Besides, long-endurant ability is very important for deep sea exploration automatically. Excepting a good battery, low power consumption sensors also can increase the endurance time. For instance, optical sensor can replace acoustic sensor, because the increased range resulting on the simultaneous use of video cameras on multiple vehicles will bring back interest to such low cost, low power consumption sensor. Moreover, this sensor can be used both to acquire relevant data and to coordinate the fleet of robots.

In each mission, the multi-robot system should march in a pattern. Therefore Pattern formation and formation control play a key role in the multi-robot system. There is one kind of structure, homogeneous structure constituted by same type of robots located at the same depth, we are very interested in. The reason is that with homogeneous structure, building a pattern is more flexible because of the same type of robots. As every robot has the same equipment and the same algorithm, their roles are totally according to the environment and their positions.

For the homogeneous structures, many pattern researches of multi-robot systems on the land can be referenced. The pattern formation includes adaptive patterns and predefined patterns. In [2], more than one pattern can be automatically generated based on gene regulatory network though all of them are similar to a circle. In [3], [4], more than one predefined patterns are introduced. The multi-robot system in [3] can evaluate and select appropriate predefined patterns automatically according to the environment. The work in [5] describes a kind of multi-robot pattern formation which is independent on the number

of robots and can be fast convergent. There are also some researches in submarine. As proposed in [6], [7], circle or ellipse formations for ocean sampling are built. According to [8], multi-robot system achieves its task based on a virtual structure triangle. In [9], it presents the algorithms of addressing the self-assembly of large teams of autonomous robotic boats into predefined patterns. However, all of the formation control methods in [2]–[9] need an external camera or GPS for providing the global positions, which is very complex to be achieved, especially in the underwater environment because of communication limits.

Without global position, pattern can be built only depending on local information. For instance, on the land, [10] and [11] build and maintain their patterns depending on the local information, even though the collision avoidance strategies in both of them are incomplete, sometimes they fail to build or maintain formation because of collision among robots. In the water, a triangle pattern is built based on the information extracted from video camera in [14].

Moreover, because of opacity of water to electromagnetic wave, GPS and high-speed radio communication are invalid [12]. Acoustic communication is the prime way for exchanging information, but the bandwidth, speed and distance are strict limited [8]. At the same time, optical communication have been achieved in short-arrange on the land [10], [11] and underwater [13], [14].

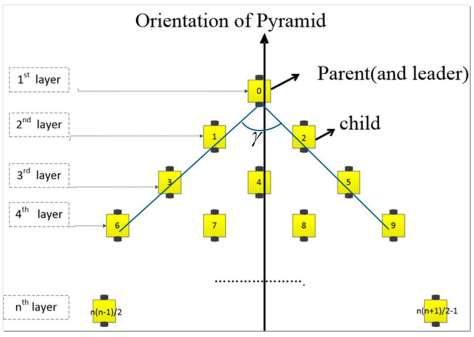
In this paper, adopting a leader-follower model [4], we design a pyramid pattern which is independent on the number of robots and adapts to a variety of tasks. With this pyramid, a control strategy for collision avoidance is proposed based on a common frame established by local information obtained from visual servoing (VS) and compass without other global sensors. Robots follow the leaders or other robots in front of them by VS, which also can reduce communication frequency and volume. Also we need an acoustic sensor to exchange some information at the beginning.

The remainder of this article is organized as follows. Section 2 presents the pyramid pattern and problem description. The formation control strategy is stated in section 3. Section 4 shows and discusses the results of first simulation experiments. Finally section 5 concludes and gives an outlook on our future work.

II. Pyramid pattern and Problem description

In our work, homogeneous structure is adopted and video cameras are the prime sensors used for information acquisition. Inspired by the swarm intelligence [17] and other literatures [3], [4], a pyramid pattern is designed with leader-follower model, shown in fig. 1. Robots can follow leader or robots located in front themselves by video cameras. And in order to strengthen constraints among robots based on graph laplacian [15] and reduce the communication distance by increasing redundancy, internal robots (such as robot 4,7,8 shown in fig.1) are introduced. The definition of pyramid pattern is:

(1) The angle γ of the entire pattern is fixed by view field of camera.



(a)

Fig. 1. Layout of the pyramidal pattern .

(2) Each robot of i_{th} layer (except the leader) has a parent located in $i-1_{th}$ layer.

(3) Distance and angle between parent and child are constant: d_{std} and θ_{std} ($\theta_{std} = \gamma/2$).

(4) Order of robot is equal to the number of robots before itself layer by layer.

We assume there are no obstacles in the environment. With this pyramid pattern, the challenge is to propose an associated formation control strategy based on local information. The local information needed in our system includes the relative positions p_{ij} (position of robot-j in robot-i frame) between robots and their neighbors evaluated from video cameras and alsolute angle θ_{ij} of robot-i when it finds robot-j measured by compass.

As a result, our goals are to propose a formation control strategy with local information to build the pattern without collision among robots in non-obstacle environment.

III. Formation control

Formation control is one key issue for multi-robot system performing tasks, which includes building and maintaining a pattern. As a common frame can be established based on local information, a control strategy based on positions is proposed. Firstly a common frame is established by exchanging the neighbors information of each robot, then all the positions of robots are translated into the common frame and the distribution of robots is obtained. Then, pyramid frame and specific pyramid (the pyramid built depending on the initial distribution of robots in each case) are designed according to the distribution of robots in the common frame. As there is no global sensor for getting the absolute positions, the robots can not know their positions in this common frame after moving. But the pyramid frame is constructed based on the common frame. In order to achieve the pyramid, trajectories need to be planned before robots move. At the same time, to prevent collision between robots, a collision avoidance algorithm (CA) is proposed so that the trajectories do not intercross. As a result, straight non-intercrossed trajectories composed by distances and orientations from initial distribution to pyramid are planned. Finally, all robots build the pyramid along their trajectories.

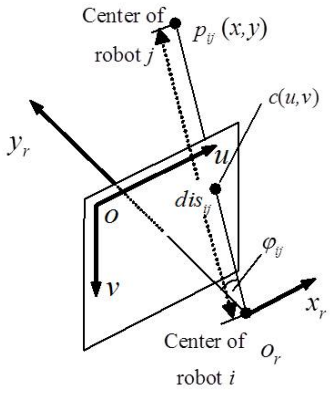


Fig. 2. camera module. The origin o_r is the optical center of camera, y_r is the optical axis of camera, and x_r is perpendicular to y_r.

All the algorithms are run similarly at every robot, in a result, the robots do not need communication after the creation of the common frame.

A. Establishing common frame

Before establishing common frame, we define two kinds of frames: camera frame and robot local frame. The y-axis of camera frame is the optical axis of camera, origin is the optical center of camera, and x-axis is perpendicular to y-axis, shown in fig. 2. The y-axis of robot local frame is along the 90 degree measured by compass and the origin is as same as that of its camera frame, shown in fig. 3.

Consequently each robot has a local frame (noted as local-i) and a camera frame (noted as camera-i). All the local-i ($i = 1, 2, ..., N$) only have translation relationships and each local-i and its camera-i only have rotation relationship.

To establish the common frame, each robot needs to detect robots in its surroundings and to get the relative positions of neighbors p_{ij} in local-i and their IDs (shown in fig. 8). To get the p_{ij} from one image, a distance module and an angle module based on camera-i are built. Assuming the distance and angle between robots are uncoupled, the distance module aims to get the relationship between the distance and the size of robot in image, while the angle module aims to obtain the relationship between the angle and the position of robot in image.

After mathematical analysis, the distance module can be expressed as a sum of 2 weighted exponential functions, shown in formula (1):

$$dis_{ij} = a * e^{-b*bh} + c * e^{-d*bh} \tag{1}$$

In which, a, b, c, d are the coefficents, bh is the height of robot-j in image of robot-i, dis_{ij} is the real distance between robot-i and robot-j.

The angle module has a linear relationship, shown as:

$$\varphi_{ij} = a * u \ + \ b \tag{2}$$

In which, a, b are the coefficents, u is the position of robot-j along the width of image of robot-i, φ_{ij} is the angle of robot-j in the camera-i of robot-i.

Then the relative positions of neighbors $p_{ij}(x_{ij}, y_{ij})$ in local-i can be calculated based on the relative distance dis_{ij} , angle of robot-j in the camera-i φ_{ij}, and the absolute angle θ_{ij} measured by compass of robot-i when seeing robot-j, shown in formula (3).

$$\begin{pmatrix} x_{ij} \\ y_{ij} \end{pmatrix} = \begin{pmatrix} \sin\theta_{ij} & \cos\theta_{ij} \\ -\cos\theta_{ij} & \sin\theta_{ij} \end{pmatrix} \begin{pmatrix} dis_{ij} * \sin\varphi_{ij} \\ dis_{ij} * \cos\varphi_{ij} \end{pmatrix} \tag{3}$$

After that, robots exchange themselves their information including the information obtained by itself and information received from other robots, the information structure is like that:

{ID of robot: i; number of its neighbors: N_i; ID of m_{th} neighbor: j; m_{th} relative position: p_{ij}}.

Here, $m \in 1, 2, ..., N_i$, N_i is the number of neighbors of robot-i.

After information merging, robot-n has the maximum neighbors, then its frame, local-n, is regarded as common frame. All the relative positions of robots calculated from formula (3) are translated to the common frame layer by layer similar to the method of traversing the depth map.

For example, the relative position p_{ni} is the position of robot-i in local-n. After local-n is chosen as the common frame, p_{ni} is noted as $(x_{common\_i}, y_{common\_i})$. Then the position of robot-j in common frame can be calculated by the relative position p_{ij} and position of robot-i $(x_{common\_i}, y_{common\_i})$, shown in formula (4).

$$\begin{pmatrix} x_{common\_j} \\ y_{common\_j} \end{pmatrix} = \begin{pmatrix} x_{ij} \\ y_{ij} \end{pmatrix} + \begin{pmatrix} x_{common\_i} \\ y_{common\_i} \end{pmatrix} \tag{4}$$

Finaly the initial distribution of robots in common frame is obtained.

B. Designing a specific pyramid

A specific pyramid needs be designed based on the initial distribution of robots after a common frame is built. And the intuitions are to identify: (1) the layer of each robot and the order of its parent; (2) the orientation of pyramid and position of leader. The orientation of pyramid is deduced from the distribution of robots. Then a pyramid frame can be constructed as the orientation of frame is the y-axis, origin is the geometrical center of all the positions in common frame. And leader of pyramid is located in the y-axis. Once the position of leader is identified, the positions of the other robots can be calculated according to the order of their parents.

1) Identifying the layer of each robot and the order of its parent: The layer of each robot and the order of its parent can be obtained based on the definition of pyramid, shown in formula (5) and (6).

$$\begin{cases} tmp = \begin{cases} \lfloor(\sqrt{1+8m})\rfloor, if \lfloor\sqrt{1+8m}\rfloor \ is \ odd \\ \lfloor(\sqrt{1+8m})\rfloor - 1, if \lfloor\sqrt{1+8m}\rfloor \ is \ even \end{cases} \\ layer_{current} = (tmp-1) * 0.5 + 1 \end{cases} \tag{5}$$

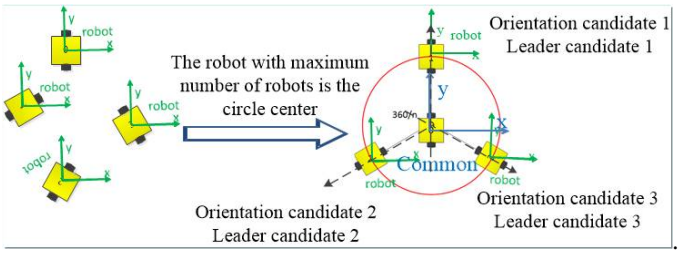


Fig. 3. Example of building pyramid frame based on CP.

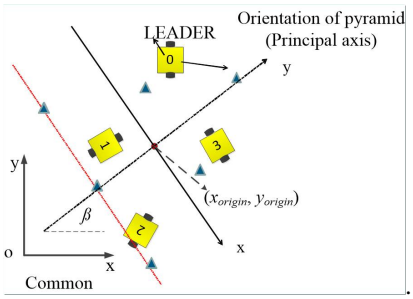


Fig. 4. Example of building pyramid frame based on PCA.

In which m is the order of current robot $Order_{itself}$.

$$Order_{parent} = \begin{cases} Order_{itself} - layer_{current}, \\ \ if\ the\ robot\ is\ the\ last\ one \\ Order_{itself} - (layer_{current} - 1) \\ else\ others \end{cases} \quad (6)$$

2) Identifying orientation of pyramid and position of leader: As the pyramid is symmetric, the distribution of robots should be separated into two sets evenly by the orientation of pyramid, which will be helpful to the collision avoidance strategy in the next part. We consider two methods to identify the orientation of pyramid: one based on an intermediate circular pattern (CP), and the other one based on principal component analysis (PCA). Then a pyramid frame is constructed.

(a) *Identification of orientation based on CP*
In this method, the robot-n with the common frame is regarded as the center robot. Then the other robots move to build the circle. The y-axis of every robot around the circle can be regarded as the orientation of pyramid, shown in fig. 3, because all of them can separate the robots into two sets evenly, and the associated robot is the leader in pyramid. In general, the robot located at the 90 degree in common frame is defaulted as the leader, and its y-axis is the orientation of the pyramid.

(b) *Identification of orientation based on PCA*
In this second method, the orientation is given by the principal axis, the eigenvector associated with the larger eigenvalue of the covariant matrix of robots positions in common frame, which is obtained from PCA [18]. Then the pyramid frame is established and pyramid is designed, the position of leader in pyramid frame is $(0, y_{leader})$. y_{leader} is decided by the minimum y-value y_{mini} (such as the y-value of robot-2 in fig. 4), the number of layers of the pyramid $layer_{all}$, d_{std} and θ_{std}: $y_{leader} = y_{mini} + layer_{all} * d_{std} * \theta_{std}$.

Then the positions of the other robots in pyramid frame are one of formula (7), which depends on the parents located in their left-front or right-front position.

$$\begin{cases} x_{child} = x_{parent} \mp d_{std} * \sin(\theta_{std}) \\ y_{child} = y_{parent} - d_{std} * \cos(\theta_{std}) \end{cases} \quad (7)$$

As the pyramid frame is constructed based on the common frame and the specific pyramid pattern is built in pyramid frame, the relationship between pyramid frame and common frame should be established for connecting the specific pyramid pattern and initial distribution of robots. An example based on PCA is shown in fig. 4.
Finally, the positions of specific pyramid in common frame are:

$$\begin{pmatrix} x_{common} \\ y_{common} \end{pmatrix} = \begin{pmatrix} \sin\beta & -\cos\beta \\ \cos\beta & \sin\beta \end{pmatrix} \begin{pmatrix} x_{pyra} \\ y_{pyra} \end{pmatrix} + \begin{pmatrix} x_{org} \\ y_{org} \end{pmatrix} \quad (8)$$

In which, β is the angle of y-axis of pyramid frame in common frame, (x_{org}, y_{org}) is the origin point of pyramid frame in common frame, and (x_{pyra}, y_{pyra}) is the position in pyramid frame obtained from equation (7).

C. Collision avoidance algorithm

In order to avoid collision among robots, we propose a collision avoidance strategy by planning non-intercrossed trajectories for them. To achieve it, the initial distribution of robots and specific pyramid need to be divided into two sets evenly for limiting their movement area. And it will be done by the orientaion of pyramid. To plan these non-intercrossed trajectories, firstly we induce a lemma for matching up the positions in initial distribution (initial group) and in specific pyramid (final group) at each set.

Lemma 1 Assume that p,f,p',f' are four distinct points in $G(N, E)$ (the graph constructed by robots, as robots are the points, communication channels between robots are the edges), shown in figure 5(a), they are separated at two groups:(p, p') and (f, f'), therefore there are two ways to link them together. One way is that $e = (p, f)$ and $e' = (p', f')$, and they are two edges in $G(N, E)$. They never cross, if the sum of their distances is smaller than the other way. The proof is given in [19].

As shown in fig. 5(a), black circles stand for positions in initial group (noted as $P_{ci}(x_{ci}, y_{ci})$), black triangles stand for positions in final group (noted as $P_{fi}(x_{fi}, y_{fi})$). The positions of both groups are in pyramid frame. We can find that for two robots, if they are pairing with the shorter distance, they will never collide with each other.
Assume each robot has one neighbor at least. According to Lemma 1, we propose a collision avoidance algorithm based on [10], [11], [14]–[16], [19], which consists of 6 steps:
(1) Positions $P_{ci}(x_{ci}, y_{ci})$ in initial group are separated into

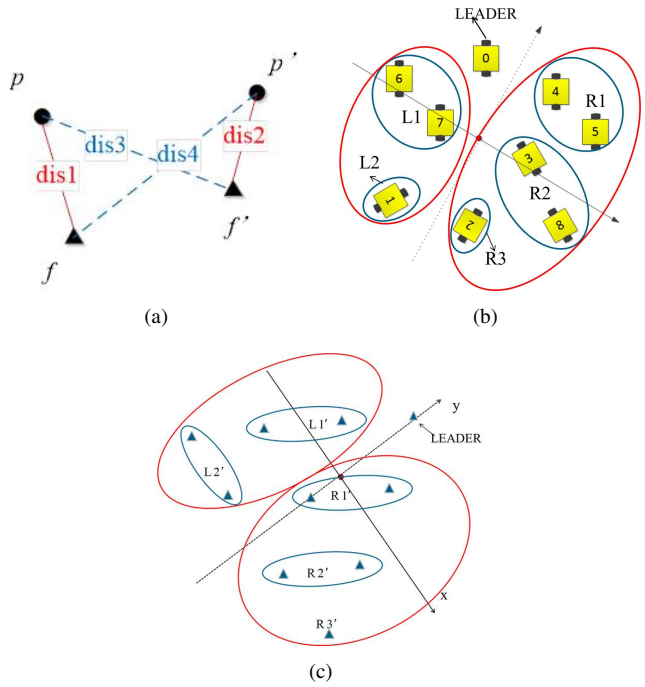


Fig. 5. Illustration of control algorithm for CA. (a) Matching up with shorter distance for CA, (b), (c) an example sketch of strategy based on PCA: (b)division of initial group and (c)division of final group in pyramid.

two sets based on the y-axis of pyramid frame, shown in fig. 5(b). In each set, all the positions are ordered according to y_{ci}.

(2) Similarly, positions $P_{fi}(x_{fi}, y_{fi})$ in final group are separated into two sets, shown in fig. 5(c). In each set, all the positions are ordered according to y_{fi}.

(3) If $y_{fi} = y_{fi+1} = ... = y_{fi+j}$, all of these positions $P_{fi}, P_{fi+1}, ...P_{fi+j}$ in each set of final group are reordered according to x_{fi}, and at the same time the positions with the same subscripts $P_{ci}, P_{ci+1}, ...P_{ci+j}$ in the associated set of initial group also are reordered according to x_{ci}.

(4) Similar to (3), if$\|y_{ci} - y_{ci+k}\| < \varepsilon, (k = 1, ..., j)$, $P_{ci}, P_{ci+1}, ...P_{ci+j}$ of each set are reordered again according to x_{ci}, and $P_{fi}, P_{fi+1}, ...P_{fi+j}$ coordinated positions in final group are also reordered according to x_{fi}. Here, $\varepsilon > (d_{max} - d_{std})(1 - \cos(\frac{\theta_{std}}{2}))$ and $\varepsilon < d_{std} cos\frac{\theta}{2}$ (shown in fig. 7(b)), in which, d_{max} is the maximum value of distances between the positions of initial and final group, d_{std} and θ_{std} are the expected distance and standard angle in pyramid respectively.

(5) $P_{ci}(x_{ci}, y_{ci})$ and $P_{fi}(x_{fi}, y_{fi})$ in each set are further separated into subgroups including one or two positions, noted as $\{\{L1, L2...\}, \{R1, R2...\}\}$, $\{\{L1', L2'...\}, \{R1', R2'...\}\}$, shown in figure 5(b) and 5(c). Then these subgroups are combining together according to their orders, called combined subgroup, for example, $L1, L1'$ compose one combined subgroup.

(6) Finally, the positions in combined subgroup match up according to lemma 1. Then straight and non-intercrossing trajectories are planned.

Proof:

1) For each combined subgroup, the robots will never collide with each other according to lemma 1.

2) For two adjacent combined subgroups, assuming the positions in initial group are $P_{c1}, P_{c2}, P_{c3}, P_{c4}$ and $y_{c1} > y_{c2} > y_{c3} > y_{c4}$; the positions in final group are $P_{f1}, P_{f2}, P_{f3}, P_{f4}$ and $y_{f1} \geq y_{f2} \geq y_{f3} \geq y_{f4}$. According to step (5), $P_{c1}, P_{c2}, P_{f1}, P_{f2}$ and $P_{c3}, P_{c4}, P_{f3}, P_{f4}$ are two combined subgroups.

(a) $y_{f1} \geq y_{f2} > y_{c3} > y_{c4}$.
These two combined subgroups have no crossing area, and all the vehicles will avoid collision with each other (figure 6(a)).

(b) $y_{c3} > y_{f1} \geq y_{f2}$, *and/or* $y_{c4} > y_{f1} \geq y_{f2}$.
If vehicles in two combined subgroups collide with each other, it means that two vehicles belonging to two combined subgroups arrive at the same position at the same time, or the final positions of one vehicle is on the path of the other vehicle, shown in figure 6(b), 6(c) respectively.
However, situation 6(b) cannot appear because of step (3). In this case, P_{f1}, P_{f2}, P_{f3} and P_{c1}, P_{c2}, P_{c3} will reordered according to their x-values. The real pairing result of 6(b) is shown in figure 7(a).
If situation 6(c) exists, P_{c1} must be located in the area Ω shown in fig. 7(b). Equation (9) is obtained in frame $o_{c4}x_{c4}y_{c4}$, because it needs to satisfy $y_{c1} > y_{c4}$ and $d_1 \leq d_4$. In other words, $0 \leq (y_{c1} - y_{c4}) \leq d_4(1 - cos\frac{\theta}{2})$, otherwise they will not collide. Therefore, if we set $\varepsilon > d_4(1 - cos\frac{\theta}{2})$, the vehicles would not collide with each other according to step (4), as $(d_{max} - d_{std}) \geq d_4$. And the real pairing result is shown in figure 7(c).

$$\Omega = \int_0^m \int_0^n [(x - d_4 sin\frac{\theta}{2})^2 + (y + d_4 cos\frac{\theta}{2})^2 - d_4^2] dx dy \tag{9}$$

Where, $m = d_4(1 - cos\frac{\theta}{2})$,$n = 2d_4 sin\frac{\theta}{2}$.

Based on the analysis above, the control strategy proposed can achieve the collision avoidance successfully in any configuration of non-obstacle situation.

The trajectories can be expressed by (α_{fi}, d_{fi}). In which, α_{fi}, d_{fi} are the angle and distance between initial position and final position in pyramid respectively. After the trajectories are planned, robots can build the pyramid.

IV. Experiments

To prove the performance of our approaches, we realized simulated experiments with several underwater robots under the Blender 2.6.9 environment [14]. The CPU of PC has 4 processors, its model name is Intel(R) Xeon(R) CPU E3-1220 V3 @ 3.1GHz. The simulation is based on the dynamic model of the CISCREA vehicle (see fig.8) [20]. Each robot is equipped with one camera and two spots located in the front of the robot, and is identifiable by a bar code composed by four cylinders with black and/or white color, noted as ID.

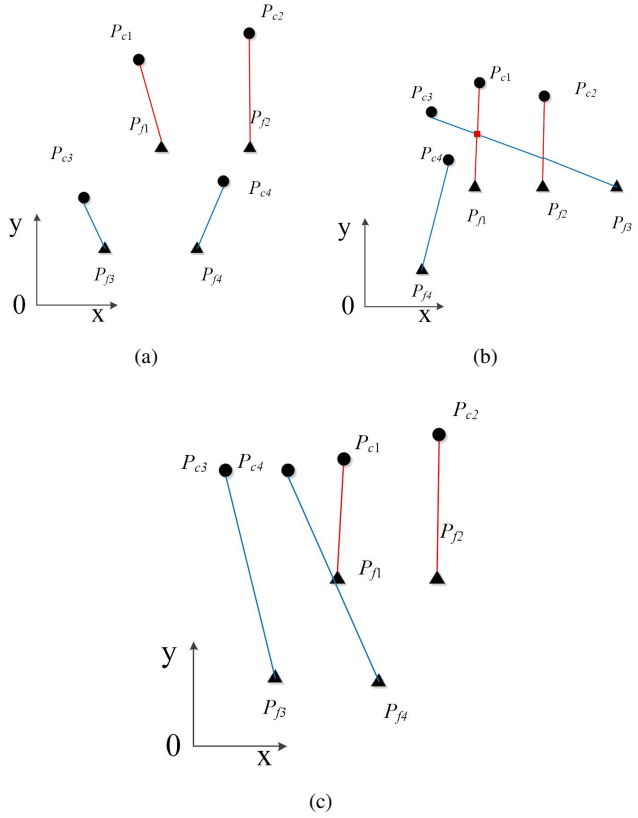


Fig. 6. Analysis of vehicles collision

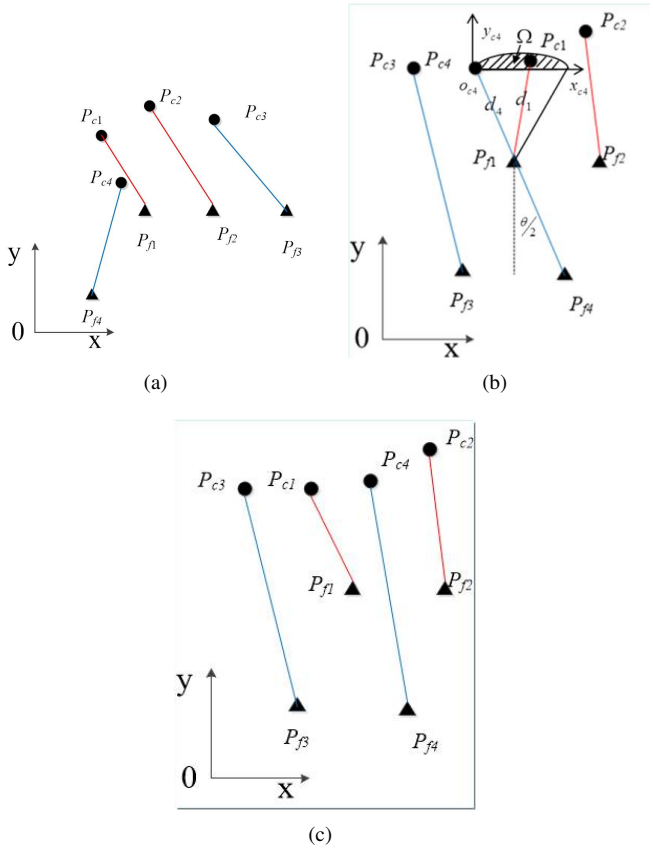


Fig. 7. Correct pairing of situations shown in figure 6

A. Calibration of distance module and angle module

In order to calibrate the coefficients in both modules (see III-A), the position and size of each robot in image need to be estimated firstly. We used preceding work done by Saad et al. [14] to detect and locate yellow parts of robots. The bar code is then decoded directly from color image according to the position of robot, shown in fig. 9.

1) Calibration of distance module: Sample data are composed by distances and sizes of yellow part of robots. At each position, distance is constant, but the size of each robot is fluctuated because of errors. In order to get accurate results, the average value of the sizes extracted from 100 continuous images is taken as the size at current position. At last, the distance module is fitted by Matlab cftool, shown in formula (10) and fig. 10.

$$dis_{ij} = 132.4 * e^{-0.1324*bh} + 4.558 * e^{-0.01497*bh} \quad (10)$$

In which, bh represents the height of the yellow part of a robot in image. To validate the fitting module, some evaluation errors are calculated: Sum of Squares for Error(SSE) is 0.1364, R-square is 0.9958, Adjusted R-square is equal to 0.9949, root-mean-square error(RMSE) is 0.09806.

2) Calibration for angle module: Similar to the sample data of distance module, at each position, angle value is unique, and u position in image is the average value of 100 times samples. Then we obtain the angle module by Matlab cftool, shown in (11) and fig. 11.

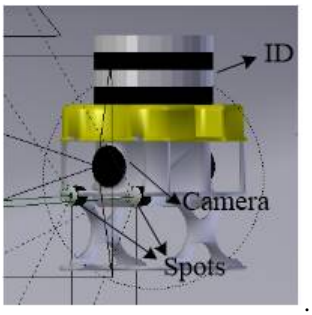


Fig. 8. Sketch of CISCREA Model.

Fig. 9. Example of image processing for extraction of positions, sizes of robots and bar code. Red rectangle expresses the position and size of robot, green area is used for bar code decoding and getting the ID of robot.

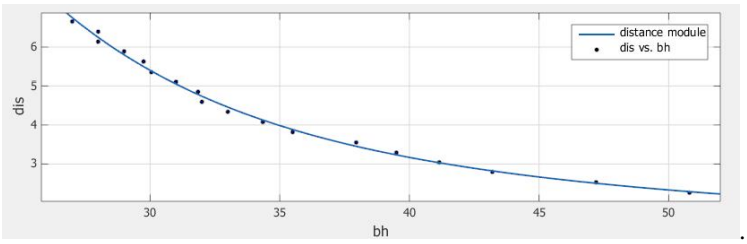


Fig. 10. Distance module. The distance between robots is exponential to the height of yellow part of robot.

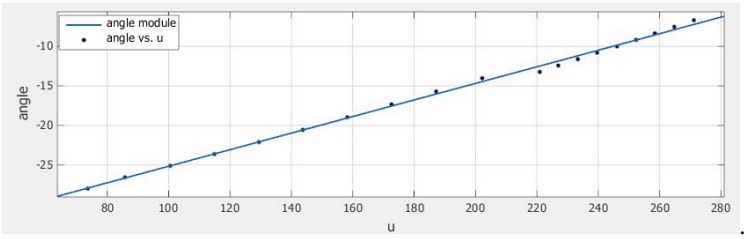


Fig. 11. Angle module. The angle between robots is linear to the u position in image.

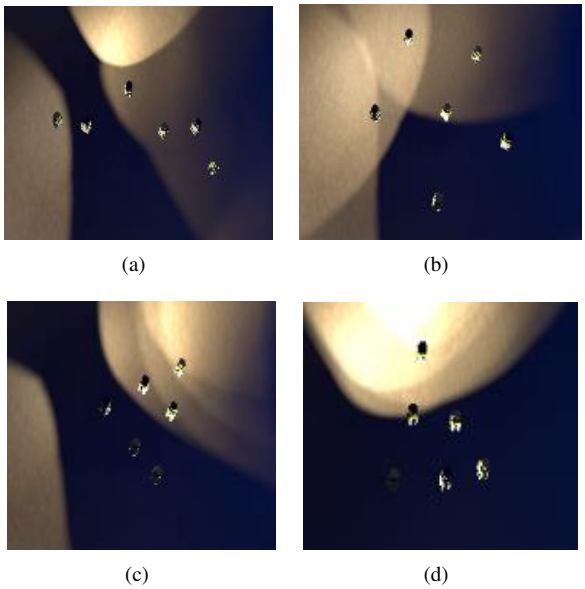


Fig. 12. Simulation of building pyramid pattern. (a) starting state of six vehicles, (b) intermediate CP from (a), and (c),(d) two pyramid patterns built, (c) based on the intermediate CP (b), and (d) according to the PCA from (a).

$$\varphi_{ij} = 0.1087 * u \ - \ 36.77 \tag{11}$$

In which, u is the position of robot along the width of image. The evaluation errors are: SSE is 0.1364, R-square is 0.9976, Adjusted R-square is equal to 0.9975, root-mean-square error (RMSE) is 0.3412.

B. Pyramid Building

With the calibration results of distance module and angle module, common frame can be established. We test the two methods, PCA and CP, to identify the orientation and leader for pyramid building. Examples of building pyramid pattern and their trajectories are shown in fig. 12 and fig. 13 with the same starting state.

The relative errors (x_err, y_err), dis_err and ang_err are estimated by comparing the actual relative positions with their parents (x_a, y_a) and expected relative positions (x_d, y_d) based on d_{std} and θ_{std} in pyramid, such as $x\_err = x_a - x_d$. And two examples, errors associated with fig. 13 based on CP and PCA are shown in fig. 14 and fig. 15 respectively.

We repeated the simulation experiments at two kinds of situations: (1) using the positions from Blender directly without image processing. (2) using the positions extracted from images. Especially, for CP, when building circle pattern, robots around the circle can fine tune their positions based on the center robot. So we tested the errors when considering the tuning step or ignoring it. The relative errors in both situations are evaluated by average errors and standard deviation of errors, noted as $aver\_*$ and $stdev\_*$. An example of relative errors is given, $x\_av\_er/x_d = \frac{1}{M}\sum_{j=1}^{M}|x\_err_j|/x_d$, M is the number of all the relative errors in multiple trials. In a similar way, all the relative errors of CP and PCA are shown in table 1 and 2 respectively. We find that: (1) The errors using the positions from Blender directly are small, which is caused by the error of Blender. (2) The errors are mainly due to image processing algorithm and uncertain factors of distance module and angle module. (3) Fining tune position when building CP can reduce the errors.

TABLE I
FINAL RELATIVE X-ERRORS, RELATIVE Y-ERRORS, RELATIVE DISTANCE ERRORS AND ANGLE ERRORS BASED ON CA

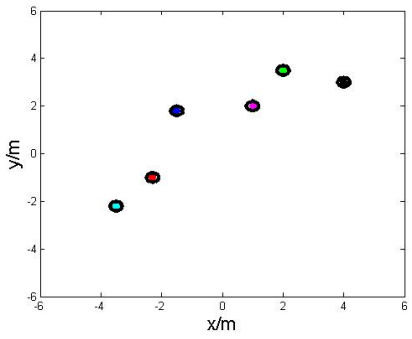
	x_av_er/ x_d	y_av_er/ y_d	dis_av_er/ d_{std}	ang_av_er/ θ_{std}
aver_no_img(%)	7.02	1.96	7.22	6.33
aver_img_no_tune(%)	30.52	11.45	16.40	33.33
aver_img_tune(%)	20.55	7.35	10.43	14.46
stdev_no_img	0.064	0.020	0.11	0.054
stdev_img_no_tune	0.28	0.10	0.15	0.29
stdev_img_tune	0.17	0.061	0.075	0.16

TABLE II
FINAL RELATIVE X-ERRORS, RELATIVE Y-ERRORS, RELATIVE DISTANCE ERRORS AND ANGLE ERRORS BASED ON PCA

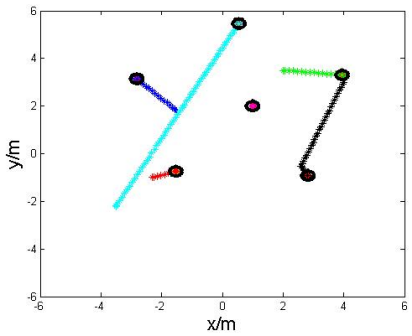
	x_err/ x_d	y_err/ y_d	dis_err/ d_{std}	angle_err/ θ_{std}
aver_no_img(%)	7.1	1.45	2.9	6.75
aver_img(%)	25.94	9.6	14.63	26.72
stdev_no_img	0.068	0.011	0.024	0.068
stdev_img_no_tune	0.31	0.087	0.11	0.27

For the time consumptions, the time of searching neighbors for both CP and PCA are the same, about 75s with one step 30 degree, the time of building the pyramid are shown in table 3. The method based on CP spends much more time because the robots fine tune their positions when building CP. The time spent in both methods fluctuates with 10% to 15% depending on the starting state.

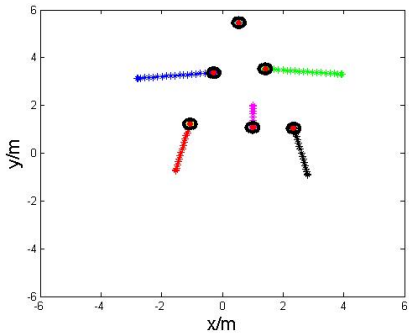
By comparison, the method based on CP provides more candidates for the orientation and leader, and is a bit more



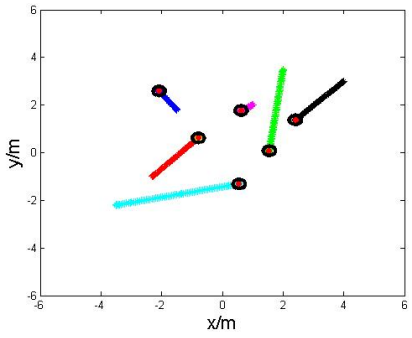
(a)



(b)

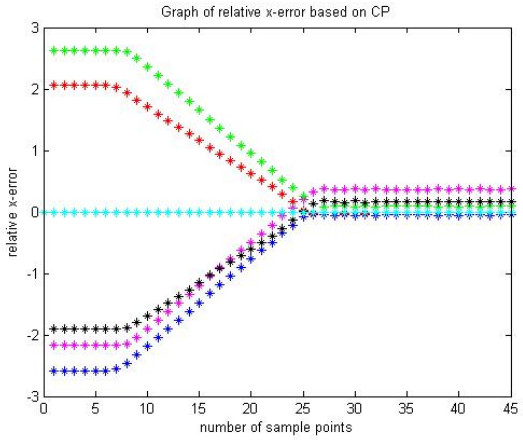


(c)

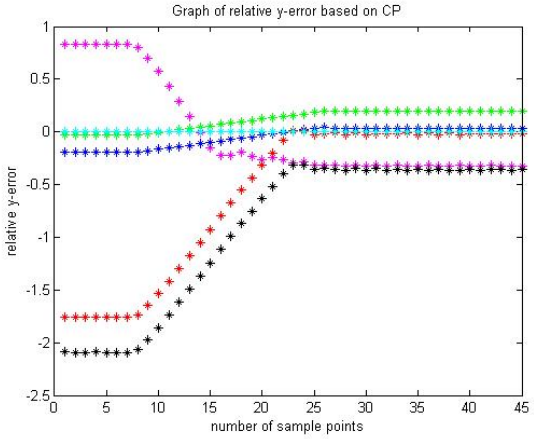


(d)

Fig. 13. Trajectories of building PP coordinating with fig.12.



(a)



(b)

Fig. 14. Relative errors between (x_a, y_a) and (x_d, y_d) based on CP: (a) relative x-errors; (b) relative y-errors. The points are sampled by taking one point every 100 points from original data.

TABLE III
TIME CONSUMPTION OF TWO WAYS OF BUILDING THE PYRAMID

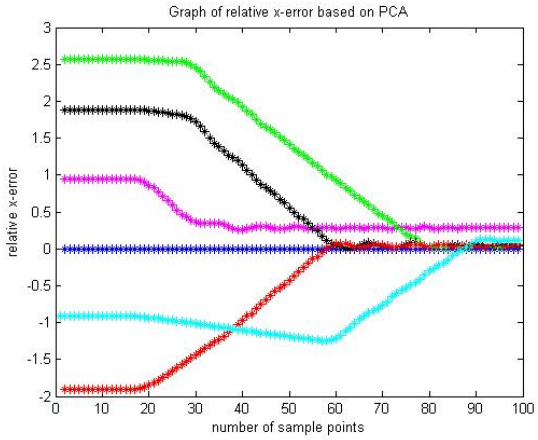
	Two ways of building the pyramid	
	based on CP/s	based on PCA/s
Time consumption	300 ± 30	100 ± 15

accurate thanks to the position fine-tuning when building circle, whereas the method based on PCA performs faster. The results show that the robots can build a formation successfully depending on local relative positions and absolute heading.

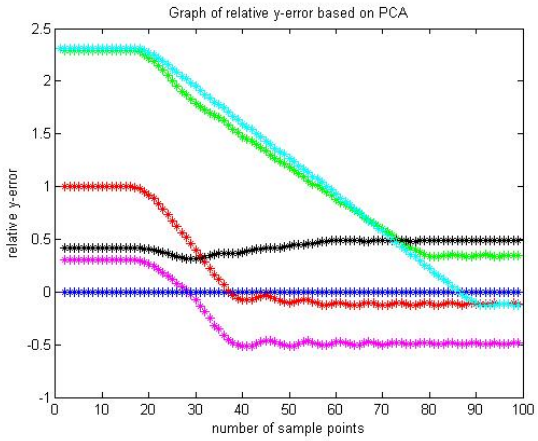
V. CONCLUSION

In this paper, a kind of multi-AUV system with homogeneous structure is presented. Visual servoing (VS) is used for information acquisition and part of communication, which reduces acoustic communication volume and frequency. Then we propose a coordination strategy: a pyramidal pattern for multi-AUV system and associated formation control strategy.

The pyramid pattern satisfies any number of robots $N(N \geq 3)$.



(a)



(b)

Fig. 15. Relative errors between (x_a, y_a) and (x_s, y_s) based on PCA: (a) relative x-errors; (b) relative y-errors. The points are sampled by taking one point every 100 points from original data.

In such a pattern, the redundant internal robots make the constraints among robots more strengthed.

A collision avoidance strategy is proposed to construct the pyramidal pattern based on the common frame built by the local information of robots from VS and compass. This collision avoidance strategy is achieved by planning straight non-intercrossed trajectories. To realize this kind of trajectories, we introduce two methods for identifying the orientation of pyramid: (1) one based on an intermediate cycle pattern; (2) the other one based on principal component analysis.

Experiments with 4 to 6 robots have done in a realistic simulation tool based on dynamic equations of real underwater robots. The results have demonstrated the performance of our coordination strategy in a non-obstacle environment.

However, there are also some problems: (1) The situations with fixed and/or unexpected obstacles need to be considered and a new formation control method should be proposed for solving this problem. (2) The distance module and angle module are calibrated in simulation environment. The modules may be different in real underwater environment because of noise and etc.. That should be re-calibrated with real robots. And Experiments of building pyramid with real robots should be done after these problems are solved.

ACKNOWLEDGMENT

The authors would like to express their great appreciation to the China Scholarship Council for their financial supports.

REFERENCES

[1] Asama H, Matsumoto A, Ishida Y. *Design of an autonomous and distributed robot system:Actress*[C]//IROS. 1989, 89: 283-290.

[2] Jin Y, Guo H, Meng Y. *A hierarchical gene regulatory network for adaptive multirobot pattern formation*[J]. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 2012, 42(3): 805-816.

[3] Yu S, Barca J C. *Autonomous formation selection for ground moving multi-robot systems*[C]//Advanced Intelligent Mechatronics (AIM), 2015 IEEE International Conference on. IEEE, 2015: 54-59.

[4] Li X, Zhu D, Qian Y. *A Survey on Formation Control Algorithms for Multi-AUV System*[J]. Unmanned Systems, 2014, 2(4): 351-359.

[5] Alonso-Mora J, Breitenmoser A, Rufli M, et al. *Multi-robot system for artistic pattern formation*[C]//Robotics and Automation (ICRA), 2011 IEEE International Conference on. IEEE, 2011: 4512-4517.

[6] Leonard N E, Paley D A, Lekien F, et al. *Collective motion, sensor networks, and ocean sampling*[J]. Proceedings of the IEEE, 2007, 95(1): 48-74.

[7] Cortes J, Martinez S, Bullo F. *Spatially-distributed coverage optimization and control with limited-range interactions*[J]. ESAIM: Control, Optimisation and Calculus of Variations, 2005, 11(4): 691-719.

[8] Fiorelli, Edward, et al. *Multi-AUV control and adaptive sampling in Monterey Bay*[J]. IEEE Journal of Oceanic Engineering 2006, 31(4): 935-948.

[9] Paulos J, Eckenstein N, Tosun T, et al. *Automated Self-Assembly of Large Maritime Structures by a Team of Robotic Boats*[J]. Automation Science and Engineering, IEEE Transactions on, 2015, 12(3): 958-968.

[10] Fredslund J., M. J. Mataric. *A general algorithm for robot formations using local sensing and minimal communication*[J]. IEEE transactions on robotics and automation 2002, 18(5): 837-846.

[11] Chiem S. Y., Cervera E. *Vision-based robot formations with Bezier trajectories*. In Eighth Conference on Intelligent Autonomous System, 2004: 191198.

[12] Stewart, W. *Remote-sensing issues for intelligent underwater systems*. In: IEEE Conference on Computer Vision and Pattern Recognition, 1991: 230-235.

[13] Davide A., Brizzolara D., Parodi G. *Building an underwater wireless sensor network based on optical: communication: research challenges and current results*. IEEE Int. Conf. on Sensor Technologies and Applications, 2009: 476479.

[14] Saad S B, Zerr B, Probst I, et al. *Hybrid Coordination Strategy of a Group of Cooperating Autonomous Underwater Vehicles*. IFAC-PapersOnLine, 2015, 48(5): 47-52.

[15] Kim Y, Mesbahi M. *On maximizing the second smallest eigenvalue of a state-dependent graph Laplacian*[J]. IEEE transactions on Automatic Control, 2006, 51(1): 116-120.

[16] Agmon N., Urieli D., Stone P. *Multiagent Patrol Generalized to Complex Environmental Conditions*.AAAI. 2011.

[17] Nighot M. K., Patil V. H., Mani G. S. *Multi-robot hunting based on swarm intelligence*[C]//Hybrid Intelligent Systems (HIS), 2012 12th International Conference on. IEEE, 2012: 203-206.

[18] Wold S., Esbensen K., Geladi P. *Principal component analysis*[J]. Chemometrics and intelligent laboratory systems, 1987, 2(1-3): 37-52.

[19] Fujinaga N., Yamauchi Y., Ono H., et al. *Pattern formation by oblivious asynchronous mobile robots*. SIAM Journal on Computing 2015, 44(3): 740-785.

[20] Yang R, Clement B., Mansour A., et al. *Modeling of a complex-shaped underwater vehicle for robust control scheme*[J]. Journal of Intelligent and Robotic Systems 2015, 80(3-4): 491-506.