Deployment of Unmanned Aerial Vehicles for Anisotropic Monitoring Tasks

Weijun Wang, *Student Member, IEEE*, Haipeng Dai, *Member, IEEE*, Chao Dong, *Member, IEEE*, Fu Xiao, *Member, IEEE*, Jiaqi Zheng, *Member, IEEE*, Xiao Cheng, Guihai Chen, *Member, IEEE*, and Xiaoming Fu, *Senior Member, IEEE*

Abstract—This paper considers the fundamental problem of deployment of Unmanned Aerial Vehicles for anisotropic monitoring Tasks (VISIT), that is, given a set of objects with determined coordinates and directions in 2D area, deploy a fixed number of UAVs by adjusting their coordinates and orientations such that the overall monitoring utility for all objects is maximized. We develop a theoretical framework to address VISIT problem. First, we establish monitoring model whose quality of monitoring (QoM is anisotropic with monitoring angle and varying with various monitoring distance. To the best of our knowledge, we are the first considering the anisotropy of monitoring angle.Then, we propose a framework consisting of area discretization and Monitoring Dominating Set (MDS) extraction to reduce the infinite solution space of VISIT to a limited one with performance bound.Finally, we model the reformulated problem as maximizing a monotone submodular function subject to a matroid constraint, and present a greedy algorithm with $1 - 1/e - \epsilon$ approximation ratio. We conduct both simulations and field experiments to evaluate our framework, and the results show that our algorithm outperforms comparison algorithms by at least 41.3 percent.

Index Terms—Unmanned aerial vehicle, anisotropy, quality of monitoring, approximation algorithm

1 Introduction

Camera sensor networks have attracted great attention in recent years as they provide detailed data of environments by retrieving rich information in the form of images and videos [1], [2]. Many applications with camera sensor networks are occurred near around us, such as surveillance, traffic monitoring, crowd protection. However, in temporary situations, such as assembly, concerts, matches, and outdoor speeches, establishing stationary camera sensor networks in advance may cost too much time and money, and may be inconvenient or even impossible. Fortunately, development of Unmanned Aerial Vehicle (UAV) technology in the past few years [3], [4], [5], [6], [7] offers a promising way to address this issue. With the low-cost and agile UAVs, camera sensor networks can be deployed dynamically to provide real-time, reliable, and high-quality images and videos. For example, *DJI Phantom 4* UAV we used in our experiments can fly at $72\,km/h$, rise at $6\,m/s$, swerve at $250\,^\circ/s$, and provide $2K$ real-time images and videos [8].

To obtain better monitoring performance, UAVs need to adjust their coordinates and orientations (the combinations of which are defined as *strategies*) to monitor frontal view of objects [9], because in most computer vision applications [10], e.g., face recognition and vehicle license plate recognition, the images from frontal view can provide key information. An object is *efficiently monitored* (defined in Section 3) by a UAV need to satisfy: First, distance between object and UAV is not too far; Second, the object is located near the orientation of the UAV as well as the UAV is located near the facing direction of the object.

Because the height of objects is much shorter than the altitude of UAVs, the models of monitoring region of a UAV and the efficient monitored region of an object can be modeled as a sector. Fig. 1 illustrates the top view of monitoring scenario (the parameters in Fig. 1 will be discussed in Section 3).

There are some papers considering the monitoring quality in camera sensor networks [11], [12], [13], [14], [15], [16], but none of them consider the continuous variable quality of monitoring in monitoring region of objects. Therefore, we introduce the anisotropic Quality of Monitoring (QoM) in our work to quantify the monitoring quality in the monitored region of the object. As shown in Fig. 2, the QoM is anisotropic and continuous. The colored surface is the image of function $QoM(distance, viewed-angle)$ and the grey-scale figure is the projection of QoM function image in the αOd plane (solid lines in grey-scale figure are QoM

- *Weijun Wang is with the State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093, China, and also with the Faculty of Mathematics and Computer Science, University of Goettingen, 37073 Göttingen, Germany. E-mail: wangalexweijun@gmail.com.*
- *Haipeng Dai is with the State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093, China, and also with Jiangsu High Technology Research Key Laboratory for Wireless Sensor Networks, Nanjing, Jiangsu 210003, China. E-mail: haipengdai@nju.edu.cn.*
- *Jiaqi Zheng, Xiao Cheng, and Guihai Chen are with the State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093, China. E-mail: {jzheng, gchen}@nju.edu.cn, gchxiao@gmail.com.*
- *Chao Dong is with the College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China. E-mail: dch999@gmail.com.*
- *Fu Xiao is with the Nanjing University of Post and Telecommunication, Nanjing 210049, China. E-mail: xiaof@njupt.edu.cn.*
- *Xiaoming Fu is with the Faculty of Mathematics and Computer Science, University of Goettingen, 37073 Göttingen, Germany. E-mail: fu@cs.uni-goettingen.de.*

Manuscript received 19 Nov. 2019; revised 7 June 2020; accepted 24 July 2020. Date of publication 29 July 2020; date of current version 7 Jan. 2022.
(Corresponding author: Guihai Chen.)
Recommended for acceptance by E. Hossain.
Digital Object Identifier no. 10.1109/TMC.2020.3012791

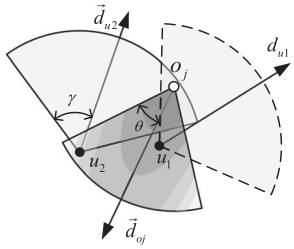


Fig. 1. Efficient monitoring model.

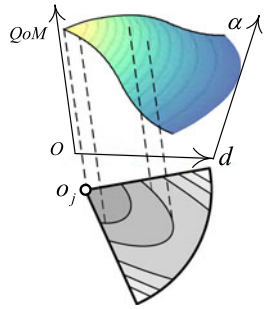


Fig. 2. Anisotropic QoM model.

contour). Given an increasing distance between object and UAV and an increasing angle between viewed direction and facing direction of object, QoM is decreasing monotonously [17], [18]. In Fig. 1, UAV u_1 and u_2 can both efficiently monitor object o_j but u_1 provides better QoM.

In computer vision, utilizing different images of objects from multi views has been proved to be dramatically improve the recognition accuracy than only from one view [9], [19], [20]. This provides us with a solid theoretical base to use multiple monitoring images of one object from different strategies to achieve better monitoring performance. We quantify this quality of one object monitored by multiple strategies as *monitoring utility*.

In this paper, we study the problem of deployment of Unmanned Aerial VehIcles for aniSotropic monItoring Tasks (VISIT). In our considered scenario, some objects are deterministically distributed at given points on the plane with known facing direction, and we have a given number of UAVs equipped with freely rotating camera to be deployed for monitoring these objects. However, because of practical factors such as limited accuracy of GPS, bias of orientation, influence of wind in the air, noise over the channel for image transmission [21], we need to consider the biased captured images. Formally, given a fixed number of UAVs and a set of objects in the plane, the VISIT problem is that considering the biased captured images to deploy the UAVs, i.e., to determine their strategies, such that the overall monitoring utility for all objects is maximized.

In previous works, there have emerged some sensor deployment methods studying efficient monitoring problem [11], [12], [13], but none of them can address our problem because they only consider binary coverage model for objects. Instead of our continuous QoM value, binary coverage model only considers an object is covered or not. Some other sensor deployment methods adopt continuous QoM models [22], [23], [24] but most of them assume omnidirectional models for both sensors and objects, thus they are also not suitable for our problem. A few sensor deployment methods use continuous QoM models and directional models for sensors and objects [25], [26], but they neglect the anisotropy of QoM, i.e., QoM varies with different monitoring angle between sensors and objects. To sum up, there is no existing method considering anisotropic QoM in camera sensor networks.

We face three main technical challenges to address the VISIT problem.

1) The practical factors, such as limited accuracy of GPS, bias of orientation, influence of wind in the air, will influence UAV capturing images. Thus, we need to address the issue of captured images with bias.
2) For each UAV, the solution space of its strategy including coordinate and orientation is infinite and continuous. Thus, we need to reduce the infinite solution space to one with limited candidate strategies.
3) VISIT is an NP-hard problem (will be proved in Section 3.4). Thus, we need to develop an approximation algorithm with performance guarantee.

To address these three challenges, we propose the corresponding method to address each of them.

1) For the first challenge, we introduce the entropy of a Gaussian random variable into our monitoring utility model and establish the conditional covariance matrix to quantify the monitoring utility with reduction of variance.
2) For the second challenge, first, we approximate the nonlinear QoM function as piecewise constant. The object function of the problem thus becomes almost linear. Then, we propose a Monitoring Dominating Set (MDS) extraction method to reduce the continuous search space of strategies of UAVs to a limited number of strategies without performance loss. Hence, the VISIT problem becomes selecting a fixed number of strategies from a set of candidate ones to maximize the overall monitoring utility, which is discrete.
3) For the third challenge, we prove that the reformulated problem falls into the scope of the problem of maximizing a monotone submodular function subject to a matroid constraint, which allows a greedy algorithm to achieve a constant approximation ratio.

We conducted both simulations and two different field experiments to evaluate our proposed framework. The results show that our algorithm outperforms comparison algorithms by at least 41.3 percent.

The rest of this paper is organized as follows. In Section 2, we review related works. Then, we present the anisotropic monitoring model and formulate the VISIT problem in Section 3. To address the VISIT problem, we present our solution in Section 4 which includes area partition in Section 4.1, MDS Extraction in Section 4.2, and problem reformulation in Section 4.3. In Sections 5 and 6, we present simulation and field experiment results. Finally, we give discussion and conclusion of this paper in Sections 7 and 8, respectively.

2 Related Work

Some sensor deployment methods consider the facing direction of objects have been proposed in [2], [12], [27], [28], [29],

[30], [31], [32], [33], [34]. In [12], Wang *et al.* proposed the full-view coverage model by introducing objects' facing direction into the coverage model. Then, the full-view coverage model is extended to more scenarios in [2], [27], [28], [29], [30], [31], [32], [33], [34]. In [28] Hu *et al.* proposed an effective algorithm to solve full-view coverage problem in the mobile heterogeneous camera sensor networks. Wang *et al.* in [27], Ma *et al.* in [29], Yu *et al.* in [30], and Liu *et al.* in [32] focused on barrier coverage with minimum number of sensors. Yu *et al.* in [30] further considered intruders' faces for most intruders' trajectories crossing the barrier and Liu *et al.* in [32] considered the mobile camera sensors. But most of them did not consider the QoM in sector coverage model.

Some sensor deployment approaches consider the QoM in their sensing region [22], [23], [24], [35], [36]. Xing *et al.* in [23] first proposed a fusion model to fuse sensing value as QoM, then utilized this fusion model to develop a deployment algorithm which needs fewer sensors than algorithms without fusion. Yang *et al.* in [24] considered the energy consumption of sensors and combined fusion model in [23] to propose energy-efficient sensor deployment algorithm. Wang *et al.* in [35] defined QoM as coverage time of objects and proposed heuristics algorithm to deploy and schedule rotatable sensor to maximize the QoM. Fusco *et al.* in [36] defined QoM as the coverage time of one object covered by different sensors and proposed a sensor selecting and orientation assigning algorithm to achieve k-cover coverage. But most of them did not consider the facing direction of objects.

A few sensor deployment approaches consider both QoM and facing direction of objects [13], [14], [15], [16], [25], [26], [37]. Tao *et al.* in [25] proposed an algorithm to solve the problem that considers the priorities of sensing quality and sensing area. Saeed *et al.* in [26] considered the size of objects and proposed an algorithm to minimize the number of cameras and guarantee there is no occlusion among objects. Wang *et al.* in [13] quantified coverage QoM as k-barrier and studied the problem of deploying a minimum number of mobile and stationary directional sensors to achieve k-barrier coverage. Li *et al.* in [14] first combined previous k-coverage QoM and full-view coverage, then proposed a k-full-view coverage algorithm to address the problem of covering fixed number of objects with a minimum number of camera sensors for a special case. Cheng *et al.* in [15] first quantified the coverage QoM as breadth of barrier-coverage and proposed an algorithm to deploy sensors covering a belt-barrier achieving β breadth, then in [16] they introduced the facing direction of objects in coverage model. However, none of them considered the anisotropic QoM in coverage area.

There are also some recent works considering the issues on autonomous cinematography with UAVs. Joubert *et al.* in [38] presented an end-to-end system for capturing well-composed photos of two subjects with UAV. They focused on addressing the issue of optimizing the locations of two subjects captured in the photos by adjusting the trajectory of UAV and camera direction. Nageli *et al.* in [39] jointly optimized 3D UAV motion plans and associated velocities. However, all these works on cinematography aim to solve the problems of taking good photos under physical limits in different scenes, which are quite different from our UAVs deployment problem.

TABLE 1
Notations

Symbol	Meaning
u_i	UAV i, or its coordinate
o_j	Object j to be monitored, or its coordinate
$\mathcal{U}$	Set of all UAVs
$\mathcal{O}$	Set of all objects
M	Number of UAVs to be deployed
N	Number of objects to be monitored
$\vec{d}_{ui}$	Orientation of UAV i
s_k	Strategy k $\langle u_i, \vec{d}_{ui}\rangle$ of UAV i with orientation $\vec{d}_{ui}$
A_l	Subarea l formed by a set of discretized sectors of O_l
$\mathcal{S}$	Set of selected strategies
γ	Monitoring angle around $\vec{d}_{ui}$
$\vec{d}_{oj}$	Orientation of object o_j
θ	Efficient angle around $\vec{d}_{oj}$ for monitoring
d_{min}	Minimum distance between UAV and object for safe
D	Monitoring distance of camera of UAV
ω	Key information in ω angle around object can be captured by a UAV
β	Distribution angle of key information of object

3 Problem Formulation

3.1 Efficient Monitoring Model

Suppose N objects $\mathcal{O} = \{o_1, o_2, \ldots, o_N\}$ are deterministically distributed on a 2D plane Ω with known coordinates o_j and orientations $\vec{d}_{oj}$. We have M UAVs $\mathcal{U} = \{u_1, u_2, \ldots, u_M\}$ equipped with cameras which can be placed anywhere on Ω with any orientation $\vec{d}_{uj}$. The tuple $\langle u_i, \vec{d}_{ui}\rangle$, saying *strategy*, where u_i denotes the coordinate of UAV i and $\vec{d}_{ui}$ denotes its orientation. By a little abuse of notation, u_i and o_j also denotes UAV i and object j. Table 1 lists the notations we used in this paper.

By incorporating the widely accepted empirical camera coverage sector model in [2], [27], [29], [30], we give the efficient monitoring definition as follows. In fact, to guarantee the safety for both objects and UAVs, we should build the covered model of object as sector ring. Namely, the UAV should not be deployed in the area with a distance less than d_{min}. However, this safe distance (1-2 m) is much shorter than Monitoring distance of UAV (30-40 m) [40], thus we still use the sector model in this paper.

Definition 3.1 (Efficient monitoring). *An object o_j is efficiently monitored if for a given vector $\vec{d}(x, y)$ (its facing direction), there is a UAV u_i, such that o_j is monitored by u_i and $\alpha(\vec{d}, \overrightarrow{o_j u_i}) \leq \theta$ (θ is called the effective angle).*

According to the camera coverage sector model and Definition 3.1, a UAV u_i with orientation $\vec{d}_{ui}$ monitors objects with non-zero QoM in a shape of a sector with *monitoring angle* γ and *radius* D, saying *monitoring area*. An object o_j with orientation $\vec{d}_{oj}$ can be efficiently monitored with non-zero QoM in a shape of a sector with *efficient angle* θ and the same *radius* D, saying *monitored area*. Fig. 1 illustrates two pairs of efficient monitoring, u_1 with o_j and u_2 with o_j.

Image resolution can be defined as the ratio of the number of pixels and the size of image, whose unit is Pixels Per Inch (PPI) [17], [18]. To obtain high QoM UAVs need to capture high-resolution images of objects. Based on the definition of resolution, within the appropriate distance between

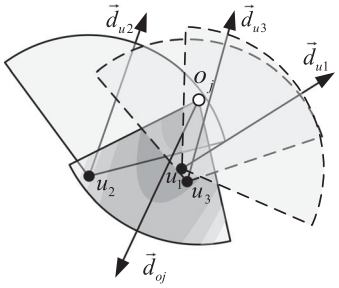


Fig. 3. Efficient monitoring model.

objects and UAVs, the closer the distance, the more pixels of objects can be captured in the image. Particularly, the number of pixels of object is inversely proportional to the square of the decrease in distance between objects and UAVs, and the number of pixels of object's frontal view is inversely proportional to the decrease in monitoring angle, i.e., $\alpha(\vec{d}_{oj}, \overrightarrow{o_j c_i})$. Therefore, the QoM of efficient monitoring pair u_i and o_j can be modeled as follows:

$$\mathcal{Q}(u_i, o_j, \vec{d}_{ui}, \vec{d}_{oj}) = \begin{cases} \frac{a}{(||u_i o_j|| + b)^2} \cos\left(\alpha(\vec{d}_{oj}, \overrightarrow{o_j u_i})\right), 0 \le ||u_i o_j|| \le D, \\ \qquad \overrightarrow{u_i o_j} \cdot \vec{d}_{ui} - \|u_i o_j\| \cos\gamma \ge 0, \\ \qquad and \quad \overrightarrow{o_j u_i} \cdot \vec{d}_{oj} - \|o_j s_i\| \cos\theta \ge 0. \\ 0, \qquad otherwise. \end{cases} \tag{1}$$

where a and b are two constants determined by the environment and the hardware, such as the electromagnetic interference from nearby base stations, the color contrast of cameras, the stability of UAV's air posture (these kinds of bias constant are also imported into the models of many other works [41], [42]). $||u_i o_j||$ denotes the distance between u_i and o_j, and $\alpha(\vec{d}_{oj}, \overrightarrow{o_j u_i})$ is the included angle between $\vec{d}_{oj}$ and $\overrightarrow{o_j u_i}$. Here using $\cos(\cdot)$ function is only for simplicity, because it represents the decreasing QoM by increasing monitoring angle. Other functions conforming to this characteristic can also work well.

3.2 Fusion Function

Multiple images from different views can provide different information for one object, thus fusing multi-viewed images can help us monitor this object better. However, fusing information is not simply linear superposition because images of an object captured from nearby strategies are often highly correlated.

As shown in Fig. 3, images captured by $\langle u_1, \vec{d}_{u1}\rangle$ and $\langle u_3, \vec{d}_{u3}\rangle$ both monitor the left side of o_j from almost the same view angle, while $\langle u_2, \vec{d}_{u2}\rangle$ monitors the right side of o_j. In other words, images captured by $\langle u_1, \vec{d}_{u1}\rangle$ is highly correlated with $\langle u_3, \vec{d}_{u3}\rangle$ but lowly correlated with $\langle u_2, \vec{d}_{u2}\rangle$. Although $\langle u_3, \vec{d}_{u3}\rangle$ alone monitors o_j better than $\langle u_2, \vec{d}_{u2}\rangle$ alone, deploying UAVs at $\langle u_1, \vec{d}_{u1}\rangle$ and $\langle u_2, \vec{d}_{u2}\rangle$ can provide more information than $\langle u_1, \vec{d}_{u1}\rangle$ and $\langle u_3, \vec{d}_{u3}\rangle$.

We use the amount of common information obtained by multiple strategies to quantify their correlation. Fig. 4a illustrates the key information distribution angle range β, the UAV extraction angle range ω, and the effective extraction angle range ω_e. The key information is distributed in the β angle range of o_j's facing direction. Each u_i can capture ω angle range of information of o_j but only the part of ω range in β captures key information, saying ω_e. β and ω are decided by types of objects of lens of cameras, thus they are different in different applications. For example, β in face recognition application is much smaller than in action recognition application because the key information is distributed on face while action can be also captured from the back view of human being. Figs. 4b and 4c give an instance. u_1 and u_2 in (b) and (c) respectively capture ω angle range information of o_j, but their common monitoring angle ω_c are different. Because ω_c in (b) is smaller than in (c), the strategies of u_1 and u_2 in (c) are more correlated. We quantify the correlation as $\frac{\omega_c}{\beta}$. If u_i monitors a set of objects $\mathcal{O}_i$ and u_j monitors a set of objects $\mathcal{O}_j$, where $\mathcal{O}_i \cap \mathcal{O}_j = \mathcal{O}_k$, then the correlation model can be expressed as

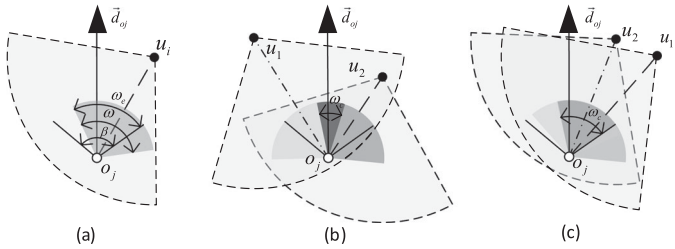


Fig. 4. Fusion model.

$$\begin{aligned} \mathcal{K}(u_i, u_j, \vec{d}_{ui}, \vec{d}_{uj}, \mathcal{O}_k) &= \frac{1}{(|\mathcal{O}_i| + |\mathcal{O}_j|)\beta} \sum_{k=1}^{|\mathcal{O}_k|} \int_{\vec{d}_{ok} - \frac{\beta}{2}}^{\vec{d}_{ok} + \frac{\beta}{2}} \omega_c d(\vec{v}) \\ &= \frac{1}{(|\mathcal{O}_i| + |\mathcal{O}_j|)\beta} \sum_{k=1}^{|\mathcal{O}_k|} (\omega - \alpha(\overrightarrow{o_k u_i}, \overrightarrow{o_k u_j})), \\ s.t. \quad & \mathcal{Q}(u_i, o_k, \vec{d}_{ui}, \vec{d}_{ok}) \ne 0, \mathcal{Q}(u_j, o_k, \vec{d}_{uj}, \vec{d}_{ok}) \ne 0, \end{aligned} \tag{2}$$

where, $|\mathcal{O}_k|$ is the size of $\mathcal{O}_k$ and $\vec{v}$ is the angle variable changing from $\vec{d}_{ok} - \frac{\beta}{2}$ to $\vec{d}_{ok} + \frac{\beta}{2}$. If there exists no same object monitored by u_i and u_j, their correlation $\mathcal{K}(\cdot) = 0$.

It needs to be mentioned that other correlation models, such as using sampling data training correlation model [43] and establishing correlation model with classic mathematical models [44], [45], are also suitable for our solution framework.

3.3 Monitoring Utility

Different subsets of strategies provide different information of objects. The more different high-quality information is captured, the better anisotropic monitoring performance can be obtained. Therefore, we need to monitor the objects not only with higher QoM but also with a lower correlation of capturing key information.

3.3.1 Fundamental of Variance Reduction

Gaussian Process (GP) is a powerful tool to illustrate our real world. An important property of GP is that given a set of random variables $\mathcal{S}$ follows GP, the joint distribution over its subset $\mathcal{A} \in \mathcal{S}$ is also Gaussian. Assuming we measure a set of data d_A corresponding to the subset $\mathcal{A}$, based on this property we can estimate the value at every point $y \in \mathcal{S}$ conditioned on these data, $P(y|\mathcal{A})$. Meanwhile, the

entropy of a Gaussian random variable y conditioned on a set of Gaussian random variables $\mathcal{A}$ can be expressed as [46]

$$H(y|\mathcal{A}) = \frac{1}{2} log\big((2\pi e)\sigma^2_{y|\mathcal{A}}\big). \tag{3}$$

It only depends on the covariance $\sigma^2_{y|\mathcal{A}}$. And according to the Probability Theory, the conditional covariance is given by:

$$\sigma^2_{y|\mathcal{A}} = \sigma^2_s - \Sigma_{y\mathcal{A}}\Sigma^{-1}_{\mathcal{A}\mathcal{A}}\Sigma_{\mathcal{A}y}, \tag{4}$$

where $\Sigma_{\mathcal{A}\mathcal{A}}$ is the covariance matrix of $\mathcal{A}$ with itself and $\Sigma_{y\mathcal{A}} = \Sigma^T_{\mathcal{A}y}$ is a row vector of the covariances of y with all variables in $\mathcal{A}$.

3.3.2 Variance Reduction

Variance reduction is a typical method for the optimal entropy problem which is widely adopted in previous work [47], [48]. Different strategies provide different reductions in the variance of the accuracy of recognition. The higher reduction of variance, the higher monitoring performance can be obtained. Moreover, according to the Entropy Theory, given a fixed covariance matrix, the conditional covariance does not depend on the actual observed values $\mathcal{A}$. This provides us with a good opportunity to place UAVs in advance.

Because the practical factors such as the limited accuracy of GPS, the bias of orientation, the influence of wind in the air and the noise over the channel for image transmission [21], the captured images are biased. We assume the aggregate effect of these factors follows Gaussian distribution which is widely accepted in many literatures [49], [50].

We can establish the kernel matrix as follows:

$$\Sigma = \begin{pmatrix} \Sigma_{\mathcal{O}\mathcal{O}} & \Sigma_{\mathcal{O}\mathcal{U}} \\ \Sigma_{\mathcal{U}\mathcal{O}} & \Sigma_{\mathcal{U}\mathcal{U}} \end{pmatrix},$$

where

$$\Sigma_{\mathcal{U}\mathcal{U}} = \begin{pmatrix} \mathcal{K}(u_1,u_1) & \mathcal{K}(u_1,u_2) & \dots & \mathcal{K}(u_1,u_m) \\ \mathcal{K}(u_2,u_1) & \mathcal{K}(u_2,u_2) & \dots & \mathcal{K}(u_2,u_m) \\ \vdots & \vdots & & \vdots \\ \mathcal{K}(u_m,u_1) & \mathcal{K}(u_m,u_2) & \dots & \mathcal{K}(u_m,u_m) \end{pmatrix}, \tag{5}$$

in which $\mathcal{K}(u_i,u_j)$ is the abbreviation of $\mathcal{K}(u_i,u_j,\vec{d}_{ui},\vec{d}_{uj},\mathcal{O}_k)$, and

$$\Sigma^T_{\mathcal{U}\mathcal{O}} = \Sigma_{\mathcal{O}\mathcal{U}} = \begin{pmatrix} \mathcal{Q}(o_1,u_1) & \mathcal{Q}(o_1,u_2) & \dots & \mathcal{Q}(o_1,u_m) \\ \mathcal{Q}(o_2,u_1) & \mathcal{Q}(o_2,u_2) & \dots & \mathcal{Q}(o_2,u_m) \\ \vdots & \vdots & & \vdots \\ \mathcal{Q}(o_n,u_1) & \mathcal{Q}(o_n,u_2) & \dots & \mathcal{Q}(o_n,u_m) \end{pmatrix}, \tag{6}$$

in which $\mathcal{Q}(o_j,u_i)$ is the abbreviation of $\mathcal{Q}(u_i,o_j,\vec{d}_{ui},\vec{d}_{oj})$. Thus, to objects set $\mathcal{O}$ and UAV set $\mathcal{U}$, the conditional covariance is

$$\sigma^2_{\mathcal{O}|\mathcal{U}} = tr(\Sigma_{\mathcal{O}\mathcal{O}}) - tr(\Sigma_{\mathcal{O}\mathcal{U}}\Sigma^{-1}_{\mathcal{U}\mathcal{U}}\Sigma_{\mathcal{U}\mathcal{O}}). \tag{7}$$

where $tr(\cdot)$ is the *trace function* of a matrix.

We stress that the assumption of Equations (1) and (2) is only one kind of metric to quantify the QoM and correlation. Generally speaking, the solution and algorithms proposed in this paper can be applied to any QoM and fusion function.

3.4 Problem Formulation

Recall our ultimate objective is to maximize the accuracy of recognition of objects and the given condition is the distribution of objects. According to the Variance Reduction, if we want to maximize the accuracy of recognition, we need to minimize the variance of $\mathcal{O}$ given $\mathcal{U}$. Namely, we need to maximize the negation of the variance. Combine the $tr(\Sigma) = tr(\Sigma_{\mathcal{O}\mathcal{O}}) + tr(\Sigma_{\mathcal{U}\mathcal{U}})$, which is

$$maximize \quad tr(\Sigma_{\mathcal{U}\mathcal{U}}) + tr(\Sigma_{\mathcal{O}\mathcal{U}}\Sigma^{-1}_{\mathcal{U}\mathcal{U}}\Sigma_{\mathcal{U}\mathcal{O}}). \tag{8}$$

Then, we define the overall monitoring utility as Equation (8), thus our task is to find the optimal strategies for all M UAVs to maximize the overall monitoring utility. With all above, the deployment of Unmanned Aerial VehIcles for aniSotropic monItoring Tasks (VISIT) problem is defined as follows:

$$(\mathbf{P1}) \quad \begin{aligned} max \quad & tr(\Sigma_{\mathcal{U}\mathcal{U}}) + tr(\Sigma_{\mathcal{O}\mathcal{U}}\Sigma^{-1}_{\mathcal{U}\mathcal{U}}\Sigma_{\mathcal{U}\mathcal{O}}), \\ s.t. \quad & |U| = M, \end{aligned}$$

where, $tr(\cdot)$ is the trace function of matrix, $\Sigma_{\mathcal{U}\mathcal{U}}$ and $\Sigma_{\mathcal{O}\mathcal{U}}$ are establish as Equations (5) and (6), and $\Sigma^{-1}_{\mathcal{U}\mathcal{U}}$ is the inverse matrix of $\Sigma_{\mathcal{U}\mathcal{U}}$. In the following theorem, we prove the VISIT problem is NP-hard.

Theorem 3.1. *The VISIT problem is NP-hard.*

4 Solution

In this section, we present an algorithm with approximation ratio $1 - 1/e - \epsilon$ to address VISIT which consists of three steps. First, because the QoM is nonlinear with distance and angle, we approximate the QoM as a piecewise constant function of distance and angle. By doing so, the monitoring region is divided into many subareas in which the distance between real QoM and approximate QoM is bounded by ϵ. Meanwhile, the approximated QoM at any point in each subarea becomes constant. Second, we present a Monitoring Dominating Set extraction method to reduce the continuous search space of strategies to a finite one without performance loss. Thereby, our problem is transformed into finding M strategies among the obtained strategies to maximize the overall monitoring utility. Third, we prove that the transformed problem falls into the realm of maximizing a monotone submodular optimization problem subject to a uniform matroid constraint, and propose a greedy algorithm to solve this problem with a performance guarantee which is $1 - 1/e$. Combine the ϵ in the first step, no performance loss in the second step, and the $1 - 1/e$ in the third step, the ultimate approximation ratio is $1 - 1/e - \epsilon$. We also prove it in Section 4.4 performance analysis.

4.1 Area Discretization

In this section, we deeply analyze the first step. We approximate the QoM as a piecewise constant function and bound the approximation error in each interval (see Theorems 4.1 and 4.2). Then the fusion function and the number subarea

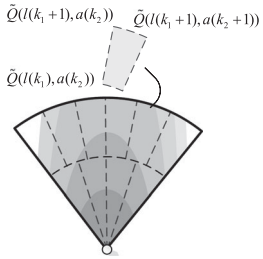


Fig. 5. Approximation illustration.

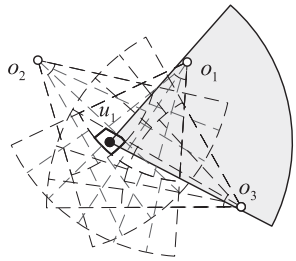


Fig. 6. Area discretization.

can be bounded by a polynomial of the same approximation error (see Theorems 4.3 and 4.4).

4.1.1 Piecewise Constant Approximation of QoM

Let $\mathcal{Q}(d,\alpha)$ denote the QoM of an object is monitored by a UAV with distance d and angle α, i.e., $\mathcal{Q}(d,\alpha)=\frac{a}{(d+b)^2}\cos\alpha$ where $0\le d\le D, 0\le\alpha\le\theta$, and $\mathcal{Q}=0$ otherwise. We use multiple piecewise constant segments $\tilde{\mathcal{Q}}(d,\alpha)$ to approximate $\mathcal{Q}(d,\alpha)$ and bound the approximation error and the computational overhead.

Fig. 5 depicts the key idea of the approximation method of $\mathcal{Q}(d,\alpha)$. Let $l(0),l(1),\ldots,l(K_1)$ be the end points in distance domain of K_1 constant segments, which divides the sector of objects into K_1 sector rings. In each sector ring, i.e. between $l(k)$ and $l(k+1)$, let $a(0),a(1),\ldots,a(K_2)$ be the end points in angle domain of constant segments, which divides each sector ring into K_2 segments both sides of $\vec{d}_{oj}$. Thus, the sector of object is divided into $K=2\sum_{j=0}^{K_1-1}K_2$ number of segments. For example, in Fig. 5, K_1 is set to 2 and K_2 for $l(0)\le d\le l(1)$ and $l(1)\le d\le l(2)$ are set to 2 and 3 respectively. Obviously, when K is larger, the less approximation error but more computational overhead is introduced.

Definition 4.1. *Setting $l(0)=0$, $l(K_1)=D$, $a(0)=0$ and $a(K_2)=\theta$, the piecewise constant QoM function $\tilde{\mathcal{K}}(d,\alpha)$ can be defined as*

$$\tilde{\mathcal{Q}}(d,\alpha)=\begin{cases}\mathcal{Q}(l(1),a(1)), & d=l(0),\alpha=a(0)\\ \mathcal{Q}(l(k),a(k)), & l(k-1)<d\le l(k),\\ & a(k-1)<\alpha\le a(k)\\ 0, & d>l(K_1),\alpha>\theta.\end{cases}\tag{9}$$

We bound the approximation error and the computational overhead with two steps. First, we regard α as a constant and bound the approximation error of $\mathcal{Q}(d,\alpha)$ in distance domain. The following theorem ensure that the approximation error in distance domain is less than ϵ_d.

Theorem 4.1. *To any given $\alpha=c$, setting $l(0)=0$, $l(K_1)=D$, and $l(k_1)=b((1+\epsilon_d)^{k_1/2}-1)$, $(k_1=1,\ldots,K_1-1$, and $K_1=\lceil\frac{ln(\mathcal{Q}(0,\alpha)/\mathcal{Q}(D,\alpha))}{ln(1+\epsilon_d)}\rceil)$, we have the approximation error:*

$$1\le\frac{\mathcal{Q}(d,c)}{\tilde{\mathcal{Q}}(d,c)}\le 1+\epsilon_d,(d\le D).\tag{10}$$

Then, we bound the approximation error in each segment. Similar, we obtain the following theorem which offers the sufficient condition to guarantee that the approximation error in each segment is less than ϵ_1.

Theorem 4.2. *In each piecewise $l(k_1)\le d\le l(k_1+1)$, setting $a(0)=0, a(K_2)=\theta$, and $a(k_2)=\arccos(\frac{1+\epsilon_d}{1+\epsilon_1})^{k_2}$, $(k_2=1,\ldots,K_2-1$, and $K_2=\lceil\frac{ln(\mathcal{Q}(l(k_1),0)/(\mathcal{Q}(l(k_1+1),\theta)\cdot(1+\epsilon_d)))}{ln((1+\epsilon_1)/(1+\epsilon_d))}\rceil)$, we have the approximation error:*

$$1\le\frac{\mathcal{Q}(d,\alpha)}{\tilde{\mathcal{Q}}(d,\alpha)}\le 1+\epsilon_1,(d\le D,\alpha\le\theta).\tag{11}$$

4.1.2 Approximation for Fusion Function

By Theorems 4.1 and 4.2, the approximation error in angle domain is bounded by $\frac{1+\epsilon_1}{1+\epsilon_d}$. Thus, the fusion function can be bounded as following theorem.

Theorem 4.3. *In each segment $a(i)\le\alpha(\vec{d}_{ok},\overrightarrow{o_ku_i})\le a(i+1)$ and $a(j)\le\alpha(\vec{d}_{ok},\overrightarrow{o_ku_j})\le a(j+1)$ in Equation (9), setting $\alpha(\vec{d}_{ok},\overrightarrow{o_ku_i})=a(i)$ and $\alpha(\vec{d}_{ok},\overrightarrow{o_ku_i})=a(j)$, then we have the approximation error:*

$$1\le\frac{\mathcal{K}(u_i,u_j,\vec{d}_{ui},\vec{d}_{uj},\mathcal{O}_k)}{\tilde{\mathcal{K}}(u_i,u_j,\vec{d}_{ui},\vec{d}_{uj},\mathcal{O}_k)}\le\frac{1+\epsilon_1}{1+\epsilon_d},(\alpha(\cdot)\le\theta).\tag{12}$$

4.1.3 Discretizing Area

In this subsection, we show how to discretize area based on piecewise constant approximation of $\mathcal{Q}(d,\alpha)$. Then, by this discretization method, we bound the solution space.

Fig. 6 illustrates the key idea of area discretization. First, we draw concentric sectors with radius $l(0),l(1),\ldots,l(K_1)$ and central angle θ centered at each object, respectively. Then, between each pair of consecutive concentric arcs $l(k_1)$ and $l(k_1+1)$, we draw line segments around both sides of orientation of objects, respectively. The extension lines of these line segments cross o_j and their included angles with $\vec{d}_{oj}$ are $a(0),a(1),\ldots,a(K_2)$, respectively. Due to geometric symmetry, if a UAV lies between two concentric sectors with radius $l(k_1)$ and $l(k_1+1)$ of an object, then this object must also lies between two sectors with the same radiuses centered at this UAV. Moreover, if the UAV monitoring this object between line segments $a(k_2)$ and $a(k_2+1)$ of this object, it will lead to a constant approximated QoM. As Fig. 6, UAV u_1 locates between sectors with radius $l(0)$ and $l(1)$ centered at objects o_1 and o_2 as well as $l(1)$ and $l(2)$ centred at object o_3, and u_1 monitors o_1 and o_3. The approximated QoM at o_1 and o_3 by u_1 is equal to $\mathcal{Q}(l(1),a(1))$ and $\mathcal{Q}(l(2),a(1))$.

Consequently, we have the following theorem.

Theorem 4.4. *The number of discretizing subarea for N objects is $O(N^2\epsilon_d^{-2}\epsilon_1^{-2})$.*

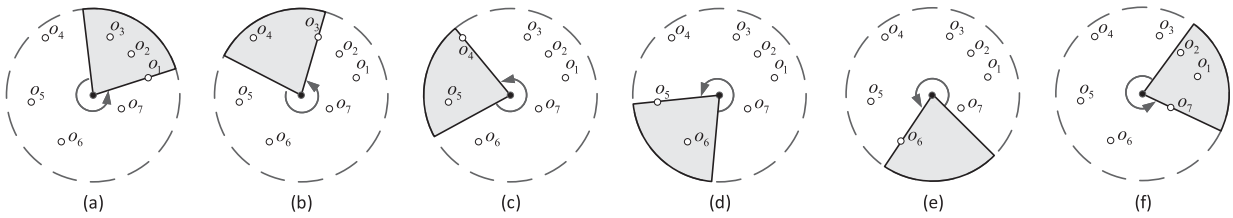


Fig. 7. An example of MDS extraction for point case.

4.2 Monitoring Dominating Set (MDS) Extraction

After the area discretization, QoM in each subarea is approximated to be constant. In this second step, our key idea is the same as the "divide and conquer" method, in which we focus on processing the relationship between objects and UAVs in one subarea and the rest subareas use the same method. We show that we just need to consider a limited number of representative monitored sets of objects rather than enumerate all possible covered sets of objects, i.e. Monitoring Dominating Sets (MDSs), and figure out these corresponding strategies. Meanwhile, we prove this step without performance loss (see Theorem 4.5).

4.2.1 Preliminaries

In the first place, we give the following definitions to assist analysis.

Definition 4.2 (Dominating Relation). *Given two strategies $\langle u_1, \vec{d}_{u1}\rangle$ and $\langle u_2, \vec{d}_{u2}\rangle$ and their monitored object sets O_1 and O_2. If $O_1 = O_2$, then we define $\langle u_1, \vec{d}_{u1}\rangle$ is equivalent to $\langle u_2, \vec{d}_{u2}\rangle$; If $O_1 \supseteq O_2$, then we define $\langle u_1, \vec{d}_{u1}\rangle$ dominates $\langle u_2, \vec{d}_{u2}\rangle$.*

Definition 4.3 (Monitoring Dominating Set). *Given a set of objects O_i monitored by a strategy $\langle u_i, \vec{d}_{ui}\rangle$, if there does not exist any strategy $\langle u_j, \vec{d}_{uj}\rangle$ such that $\langle u_j, \vec{d}_{uj}\rangle$ dominates $\langle u_i, \vec{d}_{ui}\rangle$, then O_i is a maximum dominating set and we call it Monitoring Dominating Set (MDS).*

For a given subarea, due to the limited monitoring angle of UAVs, it is possible that only some objects in the ground set of objects can be monitored by a UAV. We formally give the following definition.

Definition 4.4 (Candidate Monitored Set). *The candidate monitored set $\tilde{O}_i$ for subarea A_k are those objects that possible to be monitored by a UAV u_i with some orientation $\vec{d_{ui}}$ in A_k.*

Obviously, any MDS(s) of a subarea is a subset of the candidate monitored set of objects $\tilde{O}_i$.

Since placing UAVs monitoring MDSs is always better than monitoring its subsets, we focus on finding all MDSs as well as their strategies. In what follows, we first study a special case where a subarea is reduced to a point (point case) and then the general case (area case).

4.2.2 MDS Extraction for Point Case

Algorithm 1 clarifies the MDS extraction for point case. Basically, it is essentially a greedy algorithm which anticlockwise rotates orientation of a UAV located at the point subarea from 0 to 2π. During this process, it tracks the current set of monitoring objects, and records all MDSs. The input of this algorithm is the subarea point A_i and its candidate monitoring set $\tilde{O}_i$ and the output is all MDSs.

Fig. 7 illustrates a toy instance to show the process of MDS extraction for point case. As shown in Fig. 7a, the algorithm starts with monitoring $\{o_1\}$, then rotate UAV anticlockwise to monitor o_2 and o_3 one by one. But o_4 cannot be added into the current monitoring set otherwise $\{o_1, o_2\}$ will fall out of the monitoring region. Thus, $\{o_1, o_2, o_3\}$ is an MDS and it will be added to the collection of MDSs. Then, Algorithm 1 continues rotating the UAV and removes $\{o_1, o_2\}$ from current monitoring set and adds o_4 into it, as shown in Fig. 7b.

Algorithm 1. MDS Extraction for Point Case

Input: The subarea point A_i, the candidate monitored set $\tilde{O}_i$
Output: All MDSs

```
Set a reference ray originating from UAV as 0° and compute the angle between this reference ray and the line from the UAV to each object.
According to their angles sort all candidate monitored objects.
Initialize the orientation of the UAV to 0°.
while rotated angle is less than 360° do
    Rotate the UAV anticlockwise to monitor objects one by one until there is some monitored object will fall out of monitored.;
    if rotated angle is larger than 360° then
        terminate.
    Add the current covered set of objects to the collection of MDSs.;
    Rotate the UAV anticlockwise until a new object is included in the covered set.;
    if rotated angle is larger than 360° then
        terminate.
```

Since o_5 is beyond the current monitoring region of UAV, $\{o_3, o_4\}$ is added to the collection of MDSs. Next, the algorithm extracts MDSs of $\{o_4, o_5\}$ and $\{o_5, o_6\}$ orderly via rotating the UAV as shown in Figs. 7c and 7d. After that, Algorithm 1 removes $\{o_5\}$ from current monitoring set and try to monitor a new object, saying o_7. However, due to the limitation of monitoring angle, o_7 cannot be added in the current monitoring set as illustrated in Fig. 7e. Thus, $\{o_6\}$ is added to the collection of MDSs. The algorithm proceeds until the UAV rotates larger than 360° as depicted in Fig. 7f. Finally, the obtained collection of MDSs are $\{o_1, o_2, o_3\}, \{o_3, o_4\}, \{o_4, o_5\}$, $\{o_5, o_6\}$, $\{o_6\}$, and $\{o_7, o_1, o_2\}$.

4.2.3 MDS Extraction for Area Case

Then, we discuss the general area case and present the algorithm of MDS extraction for area case in Algorithm 2. We first give a toy instance of the process of Algorithm 2 in

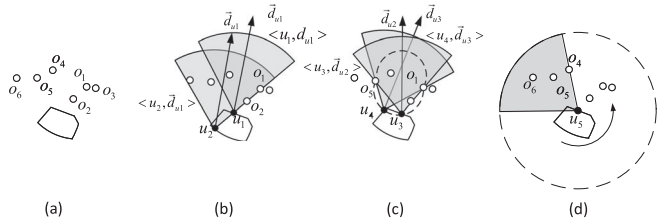


Fig. 8. An example of MDS extraction for area case.

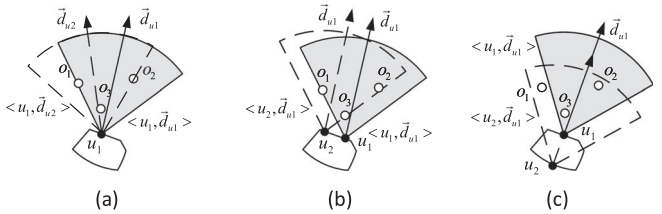


Fig. 9. Three kinds of transformation: (a) Rotation, (b) Translation, (c) Projection.

Fig. 8, then prove its correctness. As shown in Fig. 8a, suppose there are six objects in the candidate monitored set for subarea A_i. First, we draw lines crossing each pair of objects, e.g., o_1 and o_2 in Fig. 8b, and place a UAV at intersection points u_1 and u_2 with two objects lying on UAV's clockwise boundary Thus, we obtain two MDSs $\{o_1, o_2, o_4\}$ and $\{o_1, o_2, o_4, o_5, o_6\}$ as well as their strategies $\langle u_1, \vec{d}_{u1}\rangle$ and $\langle u_2, \vec{d}_{u1}\rangle$. Second, we draw arcs crossing each pair of objects with circumferential angle equals to $2 \times \gamma$, e.g., o_1 and o_5 in Fig. 8c, and place a UAV at intersection points u_3 and u_4 with two objects respectively lying on UAV's two radiuses. Thus, we obtain two MDSs $\{o_1, o_2, o_4, o_5\}$ and $\{o_1, o_4, o_5\}$ as well as their strategies $\langle u_3, \vec{d}_{u2}\rangle$ and $\langle u_4, \vec{d}_{u3}\rangle$. Third, we randomly choose a point on the boundary of subarea and execute MDS extraction algorithm for point case, as Fig. 8d. Finally, we check all the obtained MDSs and remove the MDSs which are subsets of some other MDS. In this toy instance, we reserve $\{o_1, o_2, o_4, o_5, o_6\}$ and $\{o_1, o_2, o_3, o_4, o_5\}$.

Algorithm 2. MDS Extraction for Area Case

Input: The subarea A_i and its candidate monitored set $\tilde{O}_i$
Output: MDSs and their corresponding strategies
1: **for** *all pairs of objects in* $\tilde{O}_i$*, say* o_i *and* o_j **do**
2: Draw a straight line crossing o_i and o_j, and intersect with the boundaries of subarea.;
3: Place UAVs at these intersection points and adjust their orientations, let right radius of monitoring area crossing o_i and o_j.;
4: Add the MDSs and the corresponding strategies under current setting into the solution set.;
5: Draw two arcs crossing o_i and o_j with circumferential angle $2 \cdot \gamma$ and intersect the boundaries of subarea.;
6: Place UAVs at these intersection points and adjust their orientations, let the two radiuses cross o_i and o_j respectively.;
7: Add the MDSs and the corresponding strategies under current setting into the solution set.;
8: Choose a point on the boundary of the subarea randomly and execute MDS extraction for point case algorithm and add the results into the solution set.;
9: Filter the solution set and remove the subsets of some MDSs and their corresponding strategies.

What follows, we prove the correctness of this algorithm of MDS extraction for area case.

To begin with, we give three transformations to assist our proof. As shown in Fig. 9, there are three transformations in MDS extraction for area case. All three transformations are transformed from strategy $\langle u_1, \vec{d}_{u1}\rangle$ as Fig. 9. Rotation transformation, as illustrated in Fig. 9a, keeps the coordinate of UAV u_i unchanged and rotate the orientation of UAV from $\vec{d}_{u1}$ to $\vec{d}_{u2}$. Translation transformation, as illustrated in Fig. 9b, keeps the orientation of UAV $\vec{d}_{ui}$ unchanged and move the coordinate of UAV from u_1 to u_2. Projection transformation is a special case of translation transformation, as illustrated in Fig. 9c. It keeps the orientation unchanged and moves the coordinate of UAV along the reverse direction of orientation $\vec{d}_{u1}$ until reaching some point u_2 on the boundary of subarea.

According to the transformation of projection, we have the following lemma.

Lemma 4.1. *If* $\langle u_2, \vec{d}_{u1}\rangle$ *is the projection of* $\langle u_1, \vec{d}_{u1}\rangle$*, then* $\langle u_2, \vec{d}_{u1}\rangle$ *dominates* $\langle u_1, \vec{d}_{u1}\rangle$.

Proof. As mentioned in Section 4.1, any subareas are formed by intersection of several discretized sectors of objects. Thus, any point in a subarea will not be farther than D from objects in the associated $\tilde{O}_i$ of the subarea. As a result, any objects will not get out of the sector of u_i through its arc boundary by any transformations. Moreover, any objects will not get out of the sector of u_i through its radius boundary, because during *projection* transformation the angle $\alpha(\overrightarrow{u_i o_j}, \vec{d}_{ui})$ is decreasing and will not be larger than θ as long as o_j is under monitoring before the *projection*. Therefore, *projection* transformation will not drop any originally monitored objects regarding its operated strategy, in contrast, it adds new objects to be monitored. □

Corollary 4.1. *The MDSs extracted under the case wherein UAVs located on the boundaries of a subarea dominate the MDSs extracted under the case wherein UAVs located in the whole subarea.*

By Lemma 4.1, we can get the following corollary.

Thus, we only need to consider the strategies $\langle u_i, \vec{d}_{ui}\rangle$ where u_i are on the boundaries of subarea. Now, let Γ denote the output set of strategies of Algorithm 2. We have the following theorem.

Theorem 4.5. *Given any strategy* $\langle u, \vec{d}_u\rangle$*, there exists* $\langle u_2, \vec{d}_{u2}\rangle \in \Gamma$ *such that* $\langle u_2, \vec{d}_{u2}\rangle$ *dominates* $\langle u, \vec{d}_u\rangle$.

Proof. By Corollary 4.1, we only need to consider the strategies wherein their coordinates lie on the boundaries of the subarea. Then, for an arbitrary strategy $\langle u, \vec{d}_u\rangle$, we execute the following transformations.

1) Keep the coordinate u fixed and rotate the orientation $\vec{d}_u$ anticlockwise, which is a rotation transformation as Fig. 9b, until there is at least one object, saying o_1, is going to fall out of monitoring area through the right radius. Suppose the obtained strategy is $\langle u_1, \vec{d}_{u1}\rangle$, where $u_1 = u$. Obviously, $\langle u_1, \vec{d}_{u1}\rangle$ dominates $\langle u, \vec{d}_u\rangle$.

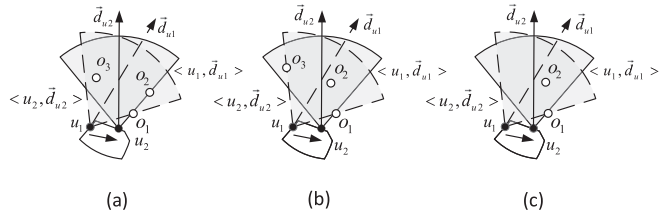


Fig. 10. Three critical conditions of transformation: (a) Two objects both touch the right radius; (b) Two objects respectively touch the left and right radius; (c) Only one object touches the boundary.

2) Keep the right radius of the UAV's monitoring area crossing o_1 and move the UAV along the boundaries of subarea, until at least another object is going to fall out of the monitoring area through either right or left radius. If no other object is going to fall out, then stop the transformation. In other words, during the process of translation and rotation transformations, any object monitored currently won't fall out of the monitoring area, formally, the newly obtained strategy $\langle u_2, \vec{d}_{u2}\rangle$ dominates $\langle u_1, \vec{d}_{u1}\rangle$.

After the above transformations, there are three possible conditions we may encounter.

1) Another object touches the right radius of the monitoring area (Fig. 10a).
2) Another object touches the left radius of the monitoring area (Fig. 10b).
3) None of other object touches the boundary of the monitoring area (Fig. 10c).

Cases 1 and 2 are critical conditions that an object which is monitored by $\langle u_1, \vec{d}_{u1}\rangle$ is going to fall out the monitoring area. While Case 3 is the situation that no objects will fall out, formally $\langle u_2, \vec{d}_{u2}\rangle$ is always equivalent to $\langle u_1, \vec{d}_{u1}\rangle$. Note that, objects won't fall out of monitoring area through the arc boundary as we have proved in Corollary 4.1.

In Algorithm 2, we can see that Step 2-4 and Step 5-7 correspond to Case 1 and 2, respectively. For Case 3, arbitrary points on the boundaries of subarea are equivalent, thus Step 8 can extract all MDSs resulted from this case. Consequently, the corresponding monitored set of objects of strategy $\langle u_2, \vec{d}_{u2}\rangle$ must be included in Γ before Step 9. Since it is dominated by some MDS in the final obtained Γ, the result follows. □

4.3 Problem Reformulation and Solution

After the second step, our problem has been reduced to a combinatorial optimization problem which is finding M strategies from a limited number of strategies extracted by MDS. In the third step, we first reformulate the problem, then prove its monotonicity and submodularity, and finally present an effective algorithm with $1-1/e$ approximation ratio to address this reformulated problem.

Let $\mathcal{S}$ be the selected set of strategies from Γ. For all possible $\mathcal{S}$ in Γ, we can compute their overall monitoring utility. The problem **P1** can be reformulated as

$$(\mathbf{P2}) \quad \begin{aligned} max \quad & tr(\Sigma_{\mathcal{SS}}) + tr(\Sigma_{\mathcal{OS}}\Sigma_{\mathcal{SS}}^{-1}\Sigma_{\mathcal{SO}}), \\ s.t. \quad & \mathcal{S} \subseteq \Gamma, |\mathcal{S}| = M, \end{aligned}$$

where $\Sigma_{\mathcal{OS}}$ and $\Sigma_{\mathcal{SS}}^{-1}$ can be easily obtained by Equation (9) of corresponding o_i and $s_i = \langle u_i, \vec{d}_{ui}\rangle$ and Equation (12) of s_i, s_j and their monitoring objects set $\mathcal{O}_i$ and $\mathcal{O}_j$.

The problem is then transformed to a combinatorial optimization problem. Now, we give the following definitions to assist further analysis before addressing **P2**.

Definition 4.5. *[51] Let S be a finite ground set. A real-valued set function $f: 2^S \to \mathbb{R}$ is normalized, monotonic, and submodular if and only if it satisfies the following conditions, respectively: (1) $f(\emptyset) = 0$; (2) $f(A \cup \{e\}) - f(A) \geq 0$ for any $A \subseteq S$ and $e \in S \backslash A$; (3) $f(A \cup \{e\}) - f(A) \geq f(B \cup \{e\}) - f(B)$ for any $A \subseteq B \subseteq S$ and $e \in S \backslash B$.*

Definition 4.6. *[51] A Matroid $\mathcal{M}$ is a strategy $\mathcal{M} = (S, L)$ where S is a finite ground set, $L \subseteq 2^S$ is a collection of independent sets, such that (1) $\emptyset \in L$; (2) if $X \subseteq Y \in L$, then $X \in L$; (3) if $X, Y \in L$, and $|X| < |Y|$, then $\exists y \in Y \backslash X$, $X \cup \{y\} \in L$.*

Definition 4.7. *[51] Given a finite set S and an integer k. A uniform matroid $\mathcal{M} = (S, L)$ is a matroid where $L = \{X \subseteq S : |X| \leq k\}$.*

Then, our problem can be reformulated as

$$(\mathbf{P3}) \quad \begin{aligned} max \quad & f(X) = tr(\Sigma_{XX}) + tr(\Sigma_{\mathcal{O}X}\Sigma_{XX}^{-1}\Sigma_{X\mathcal{O}}), \\ s.t. \quad & X \in L, \\ & L = \{X \subseteq \Gamma : |X| \leq \mathcal{M}\}. \end{aligned}$$

Lemma 4.2. *The objective function $f(X)$ is a monotone submodular function, and the constraint is a uniform matroid constraint.*

To prove Lemma 4.2, we first prove the following Claim. To save space, we skip the proof which can be found in our conference version paper [52].

Claim 4.1. *Given a* semi-positive definite (s.p.d.) *matrix Σ in a block separated form as*

$$\Sigma = \begin{pmatrix} A & E & F & G \\ E^T & B & H & I \\ F^T & H^T & C & J \\ G^T & I^T & J^T & D \end{pmatrix},$$

we have

$$\begin{aligned} 1. \quad & tr(A) + tr((EFG)(EFG)^T A^{-1}) \\ & \leq tr(A+B) + tr\left(\begin{pmatrix} F & G \\ H & I \end{pmatrix}\begin{pmatrix} F & G \\ H & I \end{pmatrix}^T \begin{pmatrix} A & E \\ E^T & B \end{pmatrix}^{-1}\right), \end{aligned} \tag{13}$$

$$\begin{aligned} 2. \quad & tr(A+B) + tr\left(\begin{pmatrix} F & G \\ H & I \end{pmatrix}\begin{pmatrix} F & G \\ H & I \end{pmatrix}^T \begin{pmatrix} A & E \\ E^T & B \end{pmatrix}^{-1}\right) \\ & - (tr(A) + tr(EFG)(EFG)^T A^{-1}) \leq \\ & tr(A+B+C) + tr\left(\begin{pmatrix} G \\ I \\ J \end{pmatrix}\begin{pmatrix} G \\ I \\ J \end{pmatrix}^T \begin{pmatrix} A & E & F \\ E^T & B & H \\ F^T & H^T & C \end{pmatrix}^{-1}\right) \\ & - \left(tr(A+C) + tr\left(\begin{pmatrix} G \\ J \end{pmatrix}\begin{pmatrix} G \\ J \end{pmatrix}^T \begin{pmatrix} A & F \\ F^T & C \end{pmatrix}^{-1}\right)\right). \end{aligned} \tag{14}$$

As inequation (13), we obtain

$$f(A \cup \{e\}) - f(A) \geq 0,$$

where, $A \subseteq \Gamma$ and element $e \in \Gamma \backslash A$. And according to (14), we have

$$(f(A \cup \{e\}) - f(A)) - (f(B \cup \{e\}) - f(B)) \geq 0,$$

where, A and B be two sets such that $A \subseteq B \subseteq \Gamma$ and element $e \in \Gamma \backslash B$.

To sum up, we obtain that $f(X)$ is a monotone submodular function.

Consequently, the reformulated problem falls into the scope of maximizing a monotone submodular function subject to a matroid constraint, which can be addressed by a greedy algorithm which achieves a good approximation [51]. Algorithm 3 shows the pseudo code of strategy selecting algorithm. In each round, Algorithm 3 greedily adds a strategy e^* to X to maximize the increment of function $f(X)$.

Algorithm 3. Strategy Selection

Input: The number of UAVs M, MDS set Γ, object function $f(X)$
Output: Strategy set X
1: $X = \emptyset$.;
2: **while** $|X| \leq M$ **do**
3: $e^* = \arg\max_{e \in \Gamma \backslash X} f(X \cup \{e\}) - f(X)$.;
4: $X = X \cup \{e^*\}$.;

4.4 Theoretical Analysis

Theorem 4.6. *The VISIT algorithm achieves an approximation ratio of* $1 - \frac{1}{e} - \epsilon$, *where* $\epsilon = 3\epsilon_1$, *and its time complexity is* $O(MN^{12}\epsilon^{-12})$.

Proof. First, we bound the approximation ratio of VISIT algorithm. Denote the two sets of strategies of all M UAVs under *OPT* (optimal solution) to problem **P1** and the reformulated problem **P2** as $\mathcal{S}_1^*$ and $\mathcal{S}_2^*$, respectively. Denote the obtained strategies of VISIT to the problem **P2** (or **P3**) as $\mathcal{S}_2$. According to the fact that a greedy algorithm of maximizing a monotone submodular function subject to a uniform matroid achieves $1 - 1/e$ approximation ratio [51], thus the approximation ratio of Algorithm 3 is $1 - 1/e$, namely

$$\begin{aligned} &tr(\Sigma_{\mathcal{S}_2\mathcal{S}_2}) + tr(\Sigma_{\mathcal{O}\mathcal{S}_2}\Sigma^{-1}_{\mathcal{S}_2\mathcal{S}_2}\Sigma_{\mathcal{S}_2\mathcal{O}}) \\ &\quad \geq \left(1 - \frac{1}{e}\right)\left(tr\left(\Sigma_{\mathcal{S}_2^*\mathcal{S}_2^*}\right) + tr\left(\Sigma_{\mathcal{O}\mathcal{S}_2^*}\Sigma^{-1}_{\mathcal{S}_2^*\mathcal{S}_2^*}\Sigma_{\mathcal{S}_2^*\mathcal{O}}\right)\right). \end{aligned} \tag{15}$$

Further, by Theorem 4.2, we have $\tilde{\mathcal{Q}}(\cdot) \geq \frac{1}{1+\epsilon_1}\mathcal{Q}(\cdot)$. By Theorem 4.3, we have $\tilde{\mathcal{K}}(\cdot) \geq \frac{1+\epsilon_d}{1+\epsilon_1}\mathcal{K}(\cdot) \geq \frac{1}{1+\epsilon_1}\mathcal{K}(\cdot)$. Thus, each entry in $\Sigma_{\mathcal{S}_2^*\mathcal{S}_2^*}, \Sigma_{\mathcal{O}\mathcal{S}_2^*}$, $\Sigma^{-1}_{\mathcal{S}_2^*\mathcal{S}_2^*}$, and $\Sigma_{\mathcal{S}_2^*\mathcal{O}}$ is approximated to the entry in problem **P1** with at most $\frac{1}{1+\epsilon_1}$ error. Then, we have

$$\begin{aligned} &tr(\Sigma_{\mathcal{S}_2^*\mathcal{S}_2^*}) + tr(\Sigma_{\mathcal{O}\mathcal{S}_2^*}\Sigma^{-1}_{\mathcal{S}_2^*\mathcal{S}_2^*}\Sigma_{\mathcal{S}_2^*\mathcal{O}}) \\ &\quad \geq \frac{1}{1+\epsilon_1} \cdot tr(\Sigma_{\mathcal{S}_1^*\mathcal{S}_1^*}) \\ &\qquad + \frac{1}{1+\epsilon_1} \cdot \frac{1}{1+\epsilon_1} \cdot \frac{1+\epsilon_d}{1+\epsilon_1} \cdot tr(\Sigma_{\mathcal{O}\mathcal{S}_1^*}\Sigma^{-1}_{\mathcal{S}_1^*\mathcal{S}_1^*}\Sigma_{\mathcal{S}_1^*\mathcal{O}}) \\ &\quad \geq \frac{1}{(1+\epsilon_1)^3} \cdot (tr(\Sigma_{\mathcal{S}_1^*\mathcal{S}_1^*}) + tr(\Sigma_{\mathcal{O}\mathcal{S}_1^*}\Sigma^{-1}_{\mathcal{S}_1^*\mathcal{S}_1^*}\Sigma_{\mathcal{S}_1^*\mathcal{O}})). \end{aligned} \tag{16}$$

Combining Inequation (15) and (16), it can be bounded as follows:

$$\begin{aligned} &tr(\Sigma_{\mathcal{S}_2\mathcal{S}_2}) + tr(\Sigma_{\mathcal{O}\mathcal{S}_2}\Sigma^{-1}_{\mathcal{S}_2\mathcal{S}_2}\Sigma_{\mathcal{S}_2\mathcal{O}}) \\ &\quad \geq \left(1 - \frac{1}{e}\right) \cdot \frac{1}{(1+\epsilon_1)^3} \cdot (tr(\Sigma_{\mathcal{S}_1^*\mathcal{S}_1^*}) + tr(\Sigma_{\mathcal{O}\mathcal{S}_1^*}\Sigma^{-1}_{\mathcal{S}_1^*\mathcal{S}_1^*}\Sigma_{\mathcal{S}_1^*\mathcal{O}})) \\ &\quad \geq \left(1 - \frac{1}{e}\right) \cdot (1-\epsilon_1)^3 \cdot (tr(\Sigma_{\mathcal{S}_1^*\mathcal{S}_1^*}) + tr(\Sigma_{\mathcal{O}\mathcal{S}_1^*}\Sigma^{-1}_{\mathcal{S}_1^*\mathcal{S}_1^*}\Sigma_{\mathcal{S}_1^*\mathcal{O}})) \\ &\quad \geq \left(1 - \frac{1}{e}\right) \cdot (1 - 3\epsilon_1 + 3\epsilon_1^2 - \epsilon_1^3) \cdot \\ &\qquad (tr(\Sigma_{\mathcal{S}_1^*\mathcal{S}_1^*}) + tr(\Sigma_{\mathcal{O}\mathcal{S}_1^*}\Sigma^{-1}_{\mathcal{S}_1^*\mathcal{S}_1^*}\Sigma_{\mathcal{S}_1^*\mathcal{O}})). \end{aligned} \tag{17}$$

For any $\epsilon_1 \leq 3$, $3\epsilon_1^2 - \epsilon_1^3 \geq 0$, we have

$$\begin{aligned} &\left(1 - \frac{1}{e}\right) \cdot (1 - 3\epsilon_1 + 3\epsilon_1^2 - \epsilon_1^3) \\ &\qquad \cdot (tr(\Sigma_{\mathcal{S}_1^*\mathcal{S}_1^*}) + tr(\Sigma_{\mathcal{O}\mathcal{S}_1^*}\Sigma^{-1}_{\mathcal{S}_1^*\mathcal{S}_1^*}\Sigma_{\mathcal{S}_1^*\mathcal{O}})) \\ &\quad \geq \left(1 - \frac{1}{e}\right) \cdot (1 - 3\epsilon_1) \cdot (tr(\Sigma_{\mathcal{S}_1^*\mathcal{S}_1^*}) + tr(\Sigma_{\mathcal{O}\mathcal{S}_1^*}\Sigma^{-1}_{\mathcal{S}_1^*\mathcal{S}_1^*}\Sigma_{\mathcal{S}_1^*\mathcal{O}})) \\ &\quad \geq \left(1 - \frac{1}{e} - 3\epsilon_1\right) \cdot (tr(\Sigma_{\mathcal{S}_1^*\mathcal{S}_1^*}) + tr(\Sigma_{\mathcal{O}\mathcal{S}_1^*}\Sigma^{-1}_{\mathcal{S}_1^*\mathcal{S}_1^*}\Sigma_{\mathcal{S}_1^*\mathcal{O}})). \end{aligned} \tag{18}$$

Therefore, by setting $\epsilon = 3\epsilon_1$, the approximation ratio of the VISIT algorithm is $1 - \frac{1}{e} - \epsilon$.

Next, we analyze the time complexity of the VISIT algorithm. First, VISIT computes the subareas by the discretization method. The complexity of this step is $O(N^2\epsilon^{-4})$ according to Theorem 4.4. Second, in each subarea MDS extracted by Algorithm 2 the combination of objects pair for Case 1 and Case 2 is $\binom{N}{2} = \frac{N(N-1)}{2}$ and each case generates $O(1)$ number of MDSs. For Case 3, its associated time complexity is also the number of MDSs in each subarea which is $O(N)$. Thus, the time complexity of Algorithm 2 (as well as the number of MDSs) is the number of subarea times the computation complexity of each subarea, i.e., $O(N^4\epsilon^{-4})$ $(= O(N^2\epsilon^{-4}) \times \max(\binom{N}{2}, O(N)))$. Third, VISIT computes correlation of each pair of strategies. Each MDS is corresponding to a strategy, thus the combination of strategies pair is $\binom{|\mathcal{S}|}{2} = \frac{|\mathcal{S}|(|\mathcal{S}|-1)}{2}$ and each selection costs $O(1)$ time. Thus, the time complexity of this step is $O(N^8\epsilon^{-8})$ $(= \binom{O(N^4\epsilon^{-4})}{2})$. At last, Algorithm 3 will perform M times loop to select M strategies. In each loop, it will multiply three matrixes generating $O(N^{12}\epsilon^{-12})$ times computation and select the best one. Thereby, the time complexity of VISIT algorithm is $O(MN^{12}\epsilon^{-12})$. □

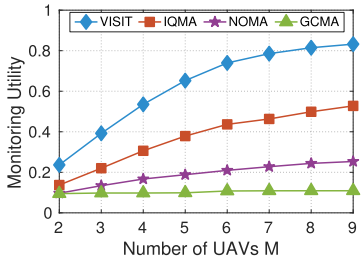


Fig. 11. M versus utility.

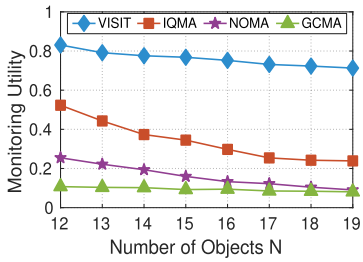


Fig. 12. N versus utility.

5 Simulation Results

5.1 Evaluation Setup

In our simulation, objects are uniformly distributed in a $10\,m \times 10\,m$ 2D square area and their orientations are randomly selected among $[0, 2\pi]$. If no otherwise stated, we set $N = 12$, $M = 9$, $\gamma = \pi/3$, $\theta = \pi/6$, $D = 3\,m$, $\omega = \pi/18$, $\beta = \pi/3$, $\epsilon = 0.2$, and $\epsilon_d = \sqrt{\epsilon/3}$, respectively. We also simulate the coordinates of UAVs which follow a 2D Gaussian distribution with both x- and y- coordinate randomly selected from a Gaussian distribution with $\mu = u_i$ and $\sigma_x = \sigma_y = 3$. Moreover, each data point in evaluation figures is computed by averaging results of 200 random topologies and normalized by dividing the best total QoM, which is $\alpha \times N$.

Since there are no existing approaches for VISIT, we present three algorithms for Comparison: (1) Grid Coordinate Monitoring Algorithm (GCMA) first divides whole 2D square area into small grids, whose side length is $D/\sqrt{2}\,m$, then extracts MDSs at each vertex of grids with MDS extraction algorithm of point case, and last greedily selects the strategies which monitors the most number of objects. In other words, GCMA doesn't consider the QoM and defines the monitoring utility of every object is just classified of monitored or not. (2) Number-Objective Monitoring Algorithm (NOMA) improves GCMA by dividing area with the intersection of monitoring sectors and extracting MDSs in each subarea with MDS extraction algorithm of area case, namely, the coordinates of UAVs are not just grid, and the objective of NOMA is maximizing the number of efficient monitored objects. (3) Isotropic-QoM Monitoring Algorithm (IQMA) improves NOMA by introducing QoM which is influenced by distance between object and UAV, but the QoM is isotropic which means it doesn't vary with different monitoring angle.

5.2 Performance Comparison

We compare the VISIT algorithm with IQMA, NOMA, and GCMA in terms of almost every parameter in Table 1. In terms of approximation ratio ϵ, the simulation results show

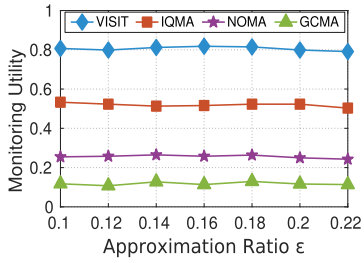


Fig. 13. ϵ versus utility.

that the monitoring utility is stable along with ϵ increases. This phenomenon let us set a relatively high ϵ to reduce the computational overhead. In addition, in terms of efficient angle θ, the monitoring utility doesn't get better along with increasing θ as we wish. The reason is that the VISIT always provides high monitoring utility with performance bound in common θ value, but if we set the extreme value of θ (i.e. 0 or 2π) the utility may change a lot. The detailed simulation results are discussed as follows.

5.2.1 Impact of Number of UAVs M

Our simulation results show that on average, VISIT outperforms IQMA, NOMA, and GCMA by 68.03 percent, 2.28 times, and 5.05 times, respectively, in terms of N. Fig. 11 shows that the monitoring utility of VISIT invariably increases with M until it approaches 1, while that of IQMA and NOMA increase to about 0.58 and 0.22, respectively, and then keep relatively stable. However, the monitoring utility of GCMA always remains low because the candidate coordinate of UAVs are limited.

5.2.2 Impact of Number of Objects N

Our simulation results show that on average, VISIT outperforms IQMA, NOMA, and GCMA by 1.24 times, 3.74 times, and 7.11 times, respectively, in terms of N. Fig. 12 shows that the monitoring utility decreases monotonically as the number of objects increases for all four algorithms. Particularly, both the monitoring utilities of GCMA decrease more slowly than that of VISIT, IQMA, and NOMA. This is because VISIT, IQMA, and NOMA select positions that can generate more monitoring utility, but GCMA cannot. Moreover, the monitoring utility of VISIT decreases more slowly than that of IQMA and NOMA when N gets larger.

5.2.3 Impact of Approximation Loss ϵ

Our simulation results show that on average, VISIT outperforms IQMA, NOMA, and GCMA by 55.18 percent, 2.15 times, and 5.83 times, respectively, in terms of ϵ. As shown in Fig. 13, the monitoring utility of VISIT fluctuates slightly when ϵ grows, but it is almost always larger than 0.8. This provides us with a good opportunity to select a larger ϵ to reduce the computation overhead of VISIT without noticeable degradation of performance.

5.2.4 Impact of Farthest Sight Distance D

Our simulation results show that on average, VISIT outperforms IQMA, NOMA, and GCMA by 57.25 percent, 2.33 times, and 5.83 times, respectively, in terms of D. Fig. 14 shows that the monitoring utility of VISIT monotonically increases with D until it

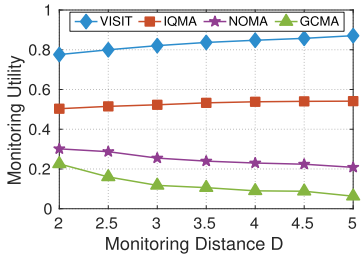


Fig. 14. D versus utility.

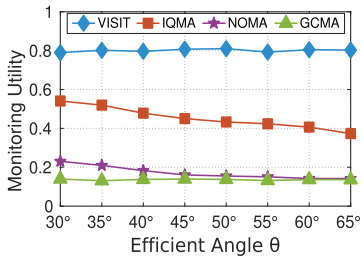


Fig. 15. θ versus utility.

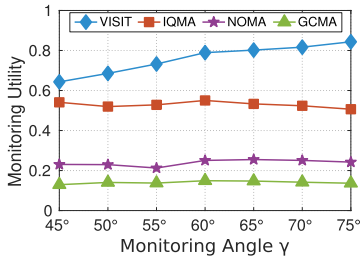


Fig. 16. γ versus utility.

approaches 1, while that of IQMA increase a little because it neglects the influence of monitoring angle. Then the monitoring utility of NOMA decreases because it only maximizes the number of monitoring objects but the QoM of additional monitoring objects in each strategy decreases when D increases. The monitoring utility of GCMA decreases because the side length of grids decreases when D increases, namely, the distance between objects and UAVs increases.

5.2.5 Impact of Efficient Angle θ

Our simulation results show that on average, VISIT outperforms IQMA, NOMA, and GCMA by 89.23 percent, 3.13 times, and 3.15 times, respectively, in terms of θ. As shown in Fig. 15, when θ grows, VISIT always maintains high monitoring utility which is almost always larger than 0.8. However, when θ grows, objects have more opportunities to be monitored. The monitoring utility of IQMA and NOMA decrease because they do not consider QoM or only consider isotropic QoM and UAVs may be placed at the strategies where more objects can be monitor but with very large monitoring angles. The monitoring utility of GCMA is always low, because UAVs can be only placed at grids.

5.2.6 Impact of Monitoring Angle γ

Our simulation results show that on average, VISIT outperforms IQMA, NOMA, and GCMA by 32.23 percent, 2.03 times, and 4.15 times, respectively, in terms of γ. Fig. 16 shows that the monitoring utility of VISIT monotonically increases with γ increasing,

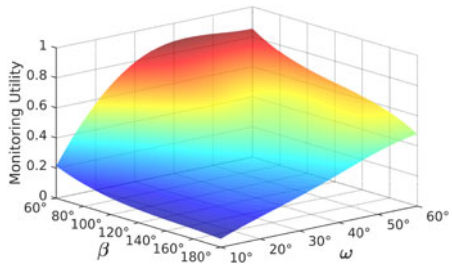


Fig. 17. β and ω versus utility.

TABLE 2
Coordinates and Orientations of Objects

Object	Coordinate (m)	Orientation
o_0	(4.32, 11.58)	234.21°
o_1	(11.93, 3.22)	114.43°
o_2	(9.94, 14.34)	256.84°
o_3	(2.44, 3.75)	133.28°
o_4	(4.40, 13.43)	294.20°
o_5	(12.50, 0.73)	103.95°
o_6	(5.56, 12.53)	168.34°
o_7	(7.62, 0.45)	240.33°
o_8	(3.55, 4.71)	197.91°
o_9	(7.09, 3.16)	282.47°
o_{10}	(3.97, 0.44)	69.88°
o_{11}	(8.97, 5.60)	194.79°
o_{12}	(10.32, 6.08)	341.20°
o_{13}	(3.28, 4.12)	114.87°
o_{14}	(8.87, 14.05)	278.88°

while the monitoring utilities of IQMA, NOMA, and GCMA almost do not change. Indeed, when γ grows, UAVs have more chance to monitoring objects, thus the monitoring utility of VISIT increases. However, IQMA, NOMA, and GCMA do not consider anisotropic QoM, thus they may place UAVs at strategies where can monitor more objects but low QoMs.

5.2.7 Impact of Object's Key Information Distribution Angle β and Captured Key Information Angle by Each UAV ω

Here we study the impact of ω and β on monitoring utility.

Fig. 17 depicts the results and each point on the surface denotes an average value of 200 experiment results. We observe that the monitoring utility increases invariably when ω increases, while monitoring utility decreases invariably when β decreases. Indeed, with a larger ω, each UAV can capture more information of objects, and with a smaller β, each object needs fewer number of UAVs to capture information.

6 Field Experiments

To verify our framework, we execute two field experiments with kinds of different objects, and thus different information of two kinds of objects and fusion functions. As shown in Fig. 19, experiment equipments consists of DJI Phantom 4 advanced UAVs and two different kinds of objects, i.e., text objects and face objects. In our experiment, 15 objects are randomly distributed in our experiment field, whose size is $15\,m \times 15\,m$, and the coordinate and orientation of objects are shown in Table 2.

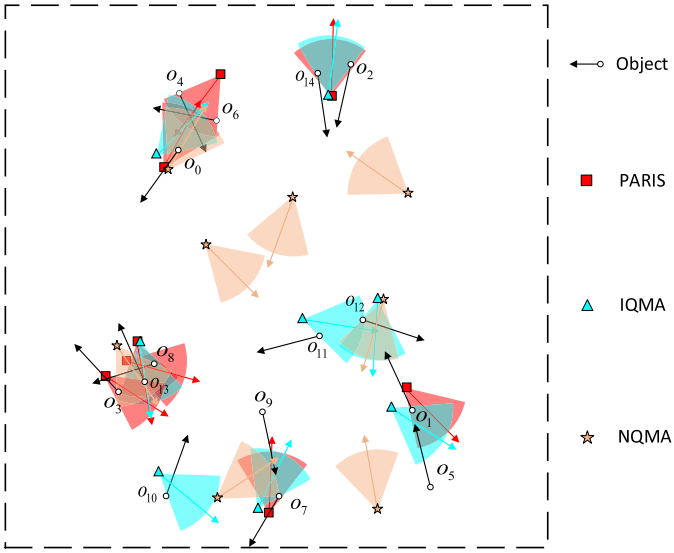


Fig. 18. Objects distribution and UAV deployment of VISIT, IQMA, and NOMA.

Fig. 19. UAV & Objects.

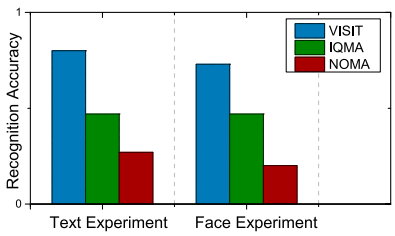


Fig. 20. Recognition results.

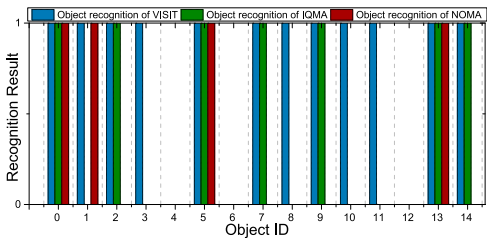


Fig. 21. Text object recognition.

Fig. 18 illustrates the distribution of 15 objects and the strategies of 8 UAVs for VISIT, IQMA, and NOMA are drawn in the red, blue, and yellow sectors respectively. Because the accuracy of GPS, the bias of orientation, the influence of wind in the air and the noise over the channel for image transmission lead to bias of pictures captured by the same strategy, we capture 5 pictures of each strategy and add them together into one picture as shown in Figs. 27, 28, and 29 and Figs. 30, 31, and 32. To keep both object and UAV safe we set isolation distance between

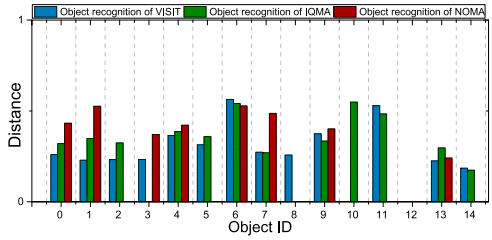


Fig. 22. Face object recognition.

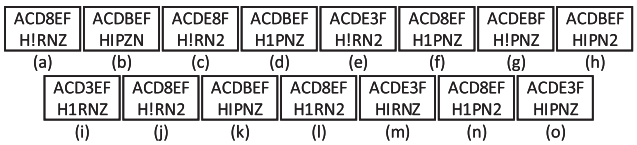


Fig. 23. Text objects.

Fig. 24. Text experiment field.

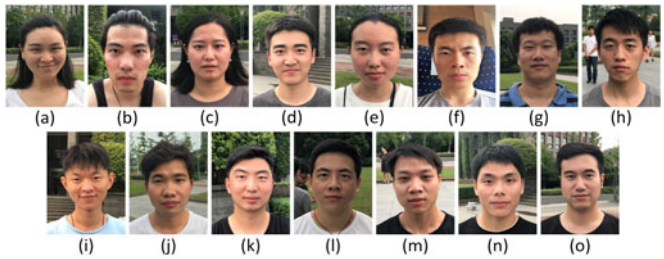


Fig. 25. Face objects.

Fig. 26. Face experiment field.

objects and UAV to 2 meters. Moreover, we set $\epsilon = 0.2$ and $\epsilon_d = \sqrt{\epsilon/3}$ in both experiments.

6.1 Text Experiment

In text experiment, we use 15 objects as shown in Fig. 19b. Each of them is made of hard distinguishable text as shown in Fig. 23 printed on an $A4$ paper and then pasted on a semicircle cylinder. Fig. 24 shows the real experiment field of text experiment which involves 15 text objects. According to the real hardware of UAV and the size of semicircle cylinder, parameters in text experiment are set to $\gamma = \pi/6$, $D = 7\,m$, $\omega = \pi/18$, $\beta = 4\pi/5$ and $\theta = 2\pi/5$.

Figs. 27, 28, and 29 show the pictures taken by the selected strategies of 8 UAVs with our algorithm VISIT and two compared algorithms IQMA and NOMA.

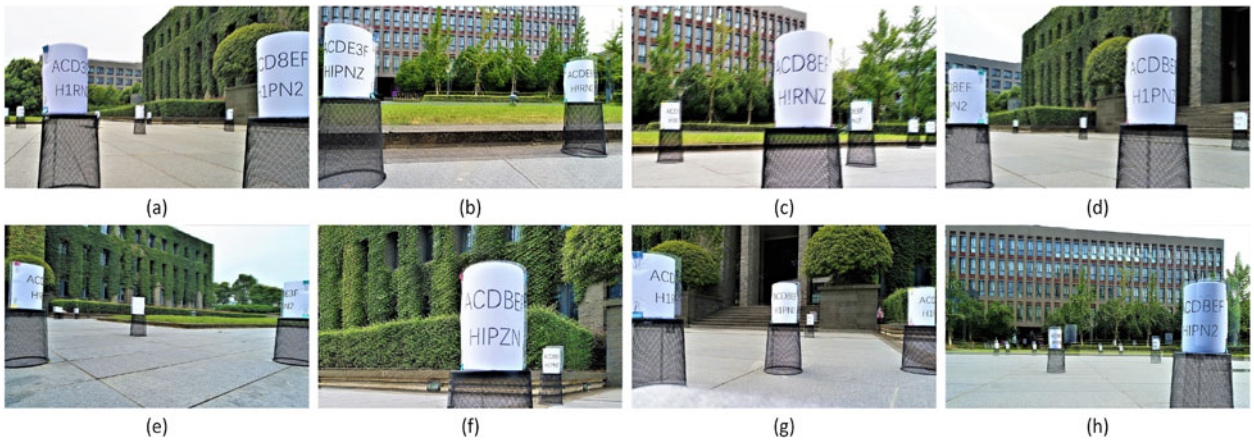


Fig. 27. Captured text pictures by VISIT.

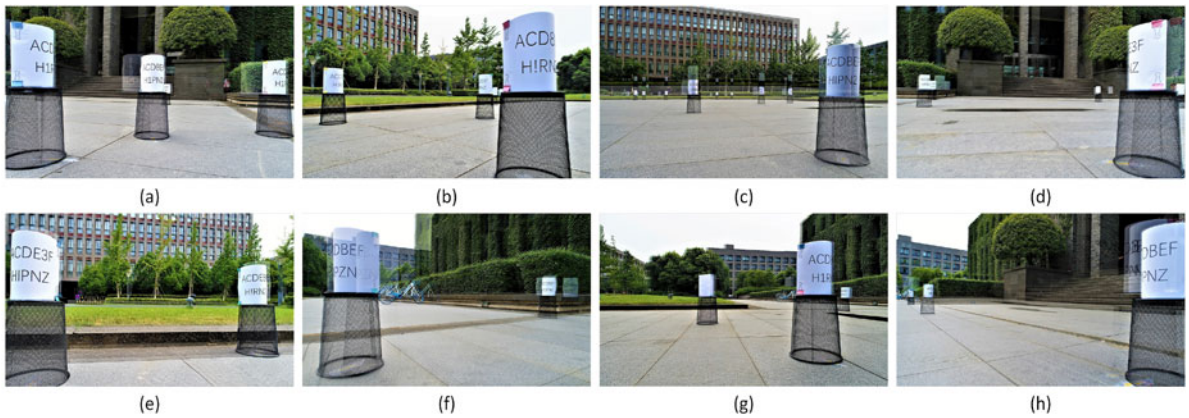


Fig. 28. Captured text pictures by IQMA.

Then, we utilize these sampling pictures of each algorithm to execute text recognition experiment. The fusion function and the text recognition algorithm we used are the existing method respectively proposed in [53] and [54]. Before executing text recognition, we also add 35 other similar texts into candidate set of recognition results. Fig. 20 illustrates the recognition accuracy of text experiment by VISIT, IQMA, and NOMA.

In text experiment, the recognition accuracies of VISIT, IQMA, and NOMA are 0.80, 0.47, and 0.27, respectively. It means VISIT outperforms IQMA and NOMA by 41.3 and 66.3 percent, respectively. Fig. 21 depicts the recognition results of each object. 1 means that the text objet has been recognized correctly and 0 means that incorrectly.

6.2 Face Experiment

In face experiment, as illustrated in Fig. 19c, we invite 15 students to attend our experiment and stand on a square, which include 3 females and 12 males whose stature are respective around $160cm - 165cm$ and $170cm - 188cm$.

15 frontal views of these students are shown in Fig. 25. Fig. 26 shows the real experiment field of face experiment which involves 15 students with known facing directions. According to the real hardware of UAV and the distribution of features in face recognition [9], parameters in face experiment are set to $\gamma = \pi/6$, $D = 7\,m$, $\omega = \pi/18$, $\beta = 2\pi/3$ and $\theta = \pi/3$.

Figs. 30, 31, and 32 show the pictures taken by the selected strategies of 8 UAVs for each of VISIT, IQMA, and NOMA. Then we use the value fusion model proposed in [53] as our fusion function and face recognition approach proposed in [55] as recognition algorithm. Similar to text recognition, we also add 85 other front view faces into candidate set of recognition results before executing face recognition experiment. Fig. 20 illustrates the recognition accuracy of by VISIT, IQMA, and NOMA.

In face experiment, the recognition accuracies for VISIT, IQMA, and NOMA are 0.73, 0.47, and 0.20, respectively, which means PARIS outperforms IQMA and NOMA by 35.6 and 72.6 percent. Fig. 22 depicts the distance of features for each object. Some bars of objects are not depicted in figure, i.e. o_{12} and o_2 of NOMA, *etc.*, because the distances of features in the sampling pictures of these objects are too large. From Fig. 22, we can see the distances of features of some object in VISIT are not better than IQMA or NOMA, i.e., o_6, o_7, o_9, o_{11}, and o_{14}, this is because the objective of our algorithm is maximizing overall monitoring utility.

7 Discussion

7.1 Deploying Minimum UAVs to Achieve a Required Monitoring Utility

The solution is nearly the same as VISIT algorithm, except that in Algorithm 3, we greedily select strategies for UAVs one by one until the required monitoring utility is achieved,

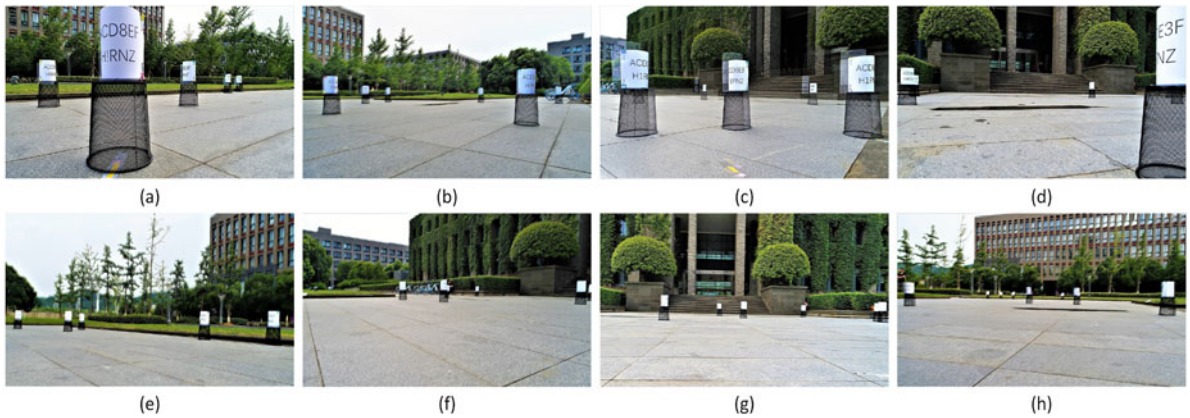


Fig. 29. Captured text pictures by NOMA.

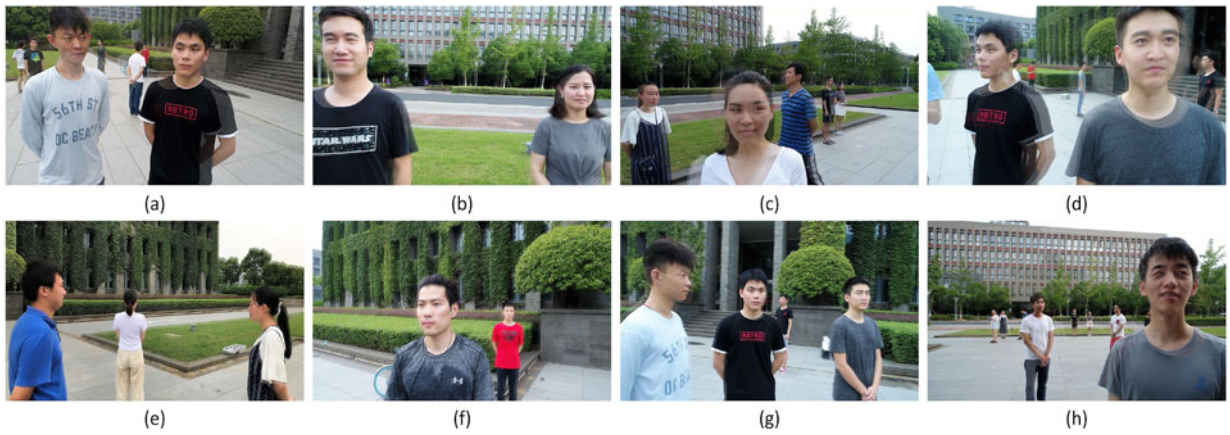


Fig. 30. Captured face pictures by VISIT.

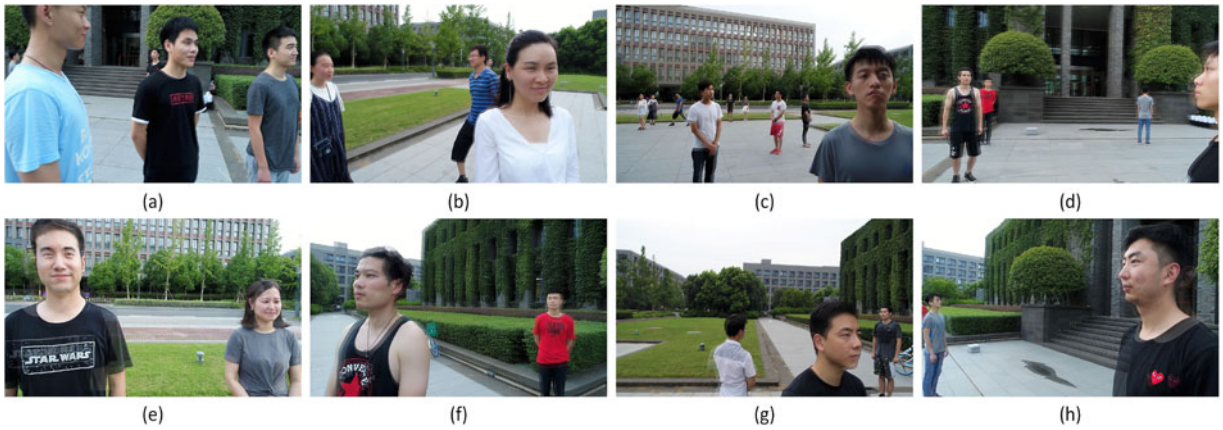


Fig. 31. Captured face pictures by IQMA.

and then output the selected strategies. According to the classical results in [56], the adapted algorithm based on VISIT achieves $\frac{1}{\ln n}$ approximation ratio, where n is the number of candidate strategies.

7.2 Heterogeneous UAVs and Objects

In some practical applications, UAVs and objects may be heterogeneous. In these application, heterogeneous UAVs can be modeled as different sectors with various γ and D. The heterogeneous objects can be described with different efficient angle θ, information distribution angle β and information captured angle ω. The VISIT problem with heterogeneous UAVs and objects can be formulated as follows. Given a set of UAVs containing Φ types of cameras, the ϕth $(1 \leq \phi \leq \Phi)$ type of camera models with parameter γ^{ϕ} and D^{ϕ}, and the number of UAVs of each type is N^{ϕ}. Given a set of objects containing Ψ types of cameras, the ψth $(1 \leq \psi \leq \Psi)$ type of camera models with parameter θ^{ψ}, β^{ψ}, ω^{ψ}, and the number of UAVs of each type is N^{ψ}. The objective function is placing these heterogeneous UAVs to maximize the overall monitoring utility for all M heterogeneous objects. According to the idea of [57], we can prove that this

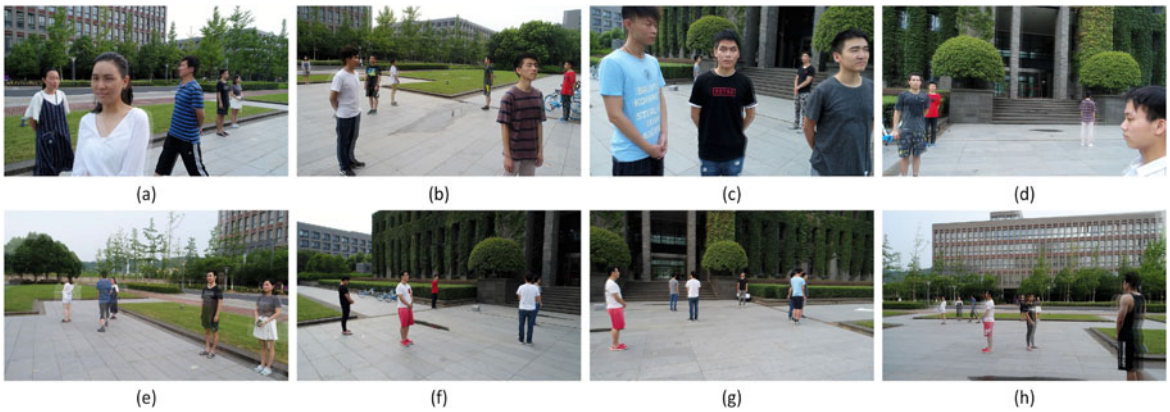


Fig. 32. Captured face pictures by NOMA.

problem falls into the realm of maximizing a submodular function subject to a partition matroid constraint, and address it using a similar greedy algorithm that achieves $\frac{1}{2}$ approximation ratio.

7.3 Practical Issues in Real

VISIT is a theoretical work giving an approximation algorithm for monitoring tasks. However, it is also an application of UAVs which stays close to our reality. Therefore, in this section, we will discuss some practical issues that may occurred in a real system.

7.3.1 Real System With VISIT Algorithm

In this section, we introduce the real experiment system we build to evaluate VISIT algorithm and discuss the practical meaning of the above mathematical theorem. In our experiment system, we serve UAVs as sampling tools that are just responsible for the figures capturing. All the computations including VISIT algorithm and face (text) recognition are executed on the server. The whole system is offline where the system can't execute real-time tasks.

Recall the input of VISIT algorithm are the number and the distribution of objects, the number of UAVs, and the parameter in approximation ratio ϵ. The outputs are the strategies of UAVs. Now we can explain the practical meaning of each theorem. Theorems 4.1, 4.2, 4.3, 4.4 explains that the iteration times of MDS extraction can be bound by a polynomial of input. Theorem 4.5 claims that MDS extraction method enumerates all possible geometry relation between UAVs and objects without omission. Theorem 4.6 proves that the approximation ratio of VISIT algorithm is bounded by a polynomial of input.

7.3.2 Considerable Volumes of Objects

In VISIT problem, we ignore both the volumes of UAVs and objects. However, in real system, we have to consider the real volumes of objects in the coverage region of UAVs. Fortunately, there are some works consider the obstruction issue in coverage problem. We can address this problem by VISIT method with a little extra preprocessing. The preprocessing for our preprocessing which includes three steps. The key idea of the preprocessing is the same as calculus. First, it discretizes the surface of each objects into λ_j number of intervals. Second, to each interval, the preprocessing method computes its norm vector and combine this interval with its norm vector to be an object. After the above two steps, we obtain $\sum_{=1}^{M} \lambda_j$ number of intervals but some of them are obstructed with each other. Third, preprocessing utilizes the method in [26] to analyze the geometry relationship among these objects and drop the obstructed objects. After the preprocessing, the rest of the objects are not obstructed, thus we can treat them as point like VISIT. Now, the problem can be reformulated to VISIT and solved with the method of this paper.

Now, we briefly analyze our method to address Considerable Volumes of Objects version VISIT problem. The denser discretization we cut the surface of one object, the more exact results we can obtain but also the more computation cost we should pay. Fortunately, we obtained the relation of them with the parameter λ_j and Theorem 4.6. Due to the limited space, we omit the detailed analysis.

7.3.3 Time Complexity Reduction

VISIT problem serves the given distribution of objects in our problem as a static state, because it slices the whole monitoring tasks into a mount of time periods. Thus, our framework is suitable for those static (offline) monitoring tasks, such as infrastructure routine check and power generator unit routine check, whose requirements of monitoring are high accuracy rather than real-time. To these tasks, UAVs are responsible to collect data and transfer to computation nodes, then powerful computation nodes process these data to get highly accurate results. Our approximation algorithm with high accuracy completely satisfies these requirements.

However, the time complexity of our algorithm is truly high for those real-time (online) monitoring tasks. Meanwhile, if the visual recognition is executing on UAVs, the power consumption is a considerable problem. To address these two issues, we need to design an energy efficient algorithm with much lesser time complexity. Fortunately, there are some techniques to reduce the time complexity, such as the correlation graph [58], [59]. In future system works, we will highlight the trade off between monitoring quality and algorithm cost time and reduce the time complexity with some techniques.

7.4 Monitoring Task of Objects and UAVs Placed in 3D Space

In this section, we consider the VISIT problem in 3D space, saying VISIT-3D. In VISIT-3D, objects and UAVs are both placed in 3D space. Specifically, some objects are

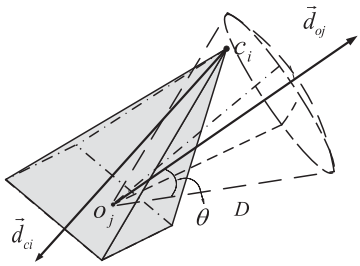


Fig. 33. Directional coverage model.

deterministically distributed at given points in the space with known facing direction, and we have a given number of UAVs to be deployed whose cameras can freely adjust their orientation. Our goal is to determine their strategies, such that the overall monitoring utility for all objects is maximized. The efficient monitoring model is illustrated in Fig. 33. UAV u_i can efficiently monitor object o_j only when it locates in the cone of object o_j (where $|\overrightarrow{u_i o_j}| \leq D$ and $\alpha(\overrightarrow{o_j u_i}, \vec{d}_{oj}) \leq \theta$) as well as o_j locates in the straight pyramid of u_i.

To solve VISIT-3D, we can also follow the three-step method of VISIT with a little adaption to adjust 3D space. First, we approximate the nonlinear QoM function as a piecewise constant function of distance and angle. Thus, each cone has been discretized into finite number of slices. These slices will intersect among each other and build a mount of subspaces. In each subspace, the QoM can be approximate to constant with performance bound. Second, in each subspace we can extract MDSs with the same geometry analysis and algorithm in our previous work [60]. Then, we obtain all the MDSs and corresponding strategies and the following step is the same as the VISIT algorithm.

8 Conclusion

In this paper, we solve the problem of deployment of Unmanned Aerial Vehicles for anisotropic monitoring tasks. The key novelty of this paper is on proposing the first algorithm for UAV deployment considering anisotropic quality of monitoring. The key contribution of this paper is: First, we build the anisotropic monitoring framework which can be used in various QoM and fusion model; Second, we develop an approximation algorithm with performance guarantee; Third, we conduct simulation and two kinds of field experiments for evaluation. The key technical depth of this paper is in reducing the infinite solution space of this optimization problem to a limited one by utilizing the techniques of area partition and Monitoring Dominating Set extraction, and modeling the reformulated problem as maximizing a monotone submodular function subject to a matroid constraint. The experiment results show that our algorithm outperforms other comparison algorithms by at least 41.3 percent.

Acknowledgments

This work was supported in part by the National Key R&D Program of China under Grant No. 2018YFB1004704, in part by the National Natural Science Foundation of China under Grant No. 61502229, 61872178, 61832005, 61672276, 61872173, 61472445, 61631020, 61702525, 61702545, 61931011, and 61321491, in part by the Natural Science Foundation of Jiangsu Province under Grant No. BK20181251, in part by the Fundamental Research Funds for the Central Universities under Grant 021014380079, in part by the Natural Science Foundation of Jiangsu Province under Grant No. BK20140076.5, in part by China Scholarships Council No. 201906190163, in part by the Fundamental Research Funds for the Central Universities under Grant 14380062, in part by the open research fund of Key Lab of Broadband Wireless Communication and Sensor Network Technology (Nanjing University of Posts and Telecommunications), Ministry of Education, in part by the Key Research and Development Project of Jiangsu Province under Grant No. BE2015154 and BE2016120, in part by the Collaborative Innovation Center of Novel Software Technology and Industrialization, Nanjing University, and in part by the Jiangsu High-level Innovation and Entrepreneurship (Shuangchuang) Program, EU H2020 COSAFE project (Grant ID: 824019).

References

[1] X. Gao, R. Yang, F. Wu, G. Chen, and J. Zhou, "Optimization of full-view barrier coverage with rotatable camera sensors," in *Proc. IEEE 37th Int. Conf. Distrib. Comput. Syst.*, 2017, pp. 870–879.

[2] S. He, D. Shin, J. Zhang, J. Chen, and Y. Sun, "Full-view area coverage in camera sensor networks: Dimension reduction and near-optimal solutions," *IEEE Trans. Veh. Technol.*, vol. 65, no. 9, pp. 7448–7461, Sep. 2016.

[3] S. Hayat, E. Yanmaz, and R. Muzaffar, "Survey on unmanned aerial vehicle networks for civil applications: A communications viewpoint," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 4, pp. 2624–2661, Fourth Quarter 2016.

[4] L. Gupta, R. Jain, and G. Vaszkun, "Survey of important issues in UAV communication networks," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 2, pp. 1123–1152, Second Quarter 2016.

[5] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Unmanned aerial vehicle with underlaid device-to-device communications: Performance and tradeoffs," *IEEE Trans. Wireless Commun.*, vol. 15, no. 6, pp. 3949–3963, Jun. 2016.

[6] Z. Qin, C. Dong, A. Li, H. Dai, Q. Wu, and A. Xu, "Trajectory planning for reconnaissance mission based on fair-energy UAVs cooperation," *IEEE Access*, vol. 7, pp. 91120–91133, 2019.

[7] Z. Qin, C. Dong, A. Li, H. Dai, Q. Wu, and A. Xu, "Fair-energy trajectory plan for reconnaissance mission based on UAVs cooperation," in *Proc. 10th Int. Conf. Wireless Commun. Signal Process.*, 2018, pp. 1–6.

[8] 2018. [Online]. Available: https://www.dji.com/phantom-4-adv/info

[9] M. Kan, S. Shan, and X. Chen, "Multi-view deep network for cross-view classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 4847–4855.

[10] H. Bay *et al.*, "SURF: Speeded up robust features," in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 404–417.

[11] H. Ma *et al.*, "On coverage problems of directional sensor networks," in *Proc. Int. Conf. Mobile Ad-Hoc Sensor Netw.*, 2005, pp. 721–731.

[12] Y. Wang and G. Cao, "On full-view coverage in camera sensor networks," in *Proc. IEEE INFOCOM*, 2011, pp. 1781–1789.

[13] Z. Wang, J. Liao, Q. Cao, H. Qi, and Z. Wang, "Achieving k-barrier coverage in hybrid directional sensor networks," *IEEE Trans. Mobile Comput.*, vol. 13, no. 7, pp. 1443–1455, Jul. 2014.

[14] C. Li, A. Rosen, and A. G. Bourgeois, "On k-full-view-coverage-algorithms in camera sensor networks," in *Proc. 2nd IEEE Int. Conf. Comput. Commun.*, 2016, pp. 2218–2222.

[15] C. F. Cheng and K. Tsai, "Distributed barrier coverage in wireless visual sensor networks with β-QoM," *IEEE Sensors J.*, vol. 12, no. 6, pp. 1726–1735, Jun. 2012.

[16] C. Cheng and K. Tsai, "Barrier coverage in wireless visual sensor networks with importance of image consideration," in *Proc. 7th Int. Conf. Ubiquitous Future Netw.*, 2015, pp. 793–798.

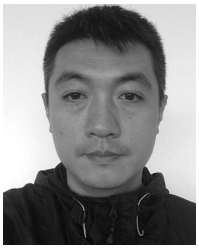
[17] A. Hamid *et al.*, *Multi-Camera Networks*. Amsterdam, The Netherlands: Elsevier, 2009.

[18] C. Shen, C. Zhang, and S. Fels, "A multi-camera surveillance system that estimates quality-of-view measurement," in *Proc. IEEE Int. Conf. Image Process.*, 2007, pp. III-193–III-196.

[19] V. Blanz, P. Grother, P. J. Phillips, and T. Vetter, "Face recognition based on frontal views generated from non-frontal images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2005, pp. 454–461.
[20] M. Kan, S. Shan, H. Zhang, S. Lao, and X. Chen, "Multi-view discriminant analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 1, pp. 188–194, Jan. 2016.
[21] 2018. [Online]. Available: https://www.dji.com/phantom-4-adv/info#specs
[22] E. Onur *et al.*, "How many sensors for an acceptable breach detection probability?" *Elsevier Comput. Commun.*, vol. 29, no. 2, pp. 173–182, 2006.
[23] G. Xing *et al.*, "Data fusion improves the coverage of wireless sensor networks," in *Proc. 15th Annu. Int. Conf. Mobile Comput. Netw.*, 2009, pp. 157–168.
[24] Q. Yang, S. He, J. Li, J. Chen, and Y. Sun, "Energy-efficient probabilistic area coverage in wireless sensor networks," *IEEE Trans. Veh. Technol.*, vol. 64, no. 1, pp. 367–377, Jan. 2015.
[25] J. Tao, T. Zhai, H. Wu, Y. Xu, and Y. Dong, "A quality-enhancing coverage scheme for camera sensor networks," in *Proc. 43rd Annu. Conf. IEEE Ind. Electron. Soc.*, 2017, pp. 8458–8463.
[26] A. Saeed, A. Abdelkader, M. Khan, A. Neishaboori, K. A. Harras, and A. Mohamed, "Argus: Realistic target coverage by drones," in *Proc. 16th ACM/IEEE Int. Conf. Inf. Process. Sensor Netw.*, 2017, pp. 155–166.
[27] Y. Wang *et al.*, "Barrier coverage in camera sensor networks," in *Proc. 12th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2011, pp. 1–10.
[28] Y. Hu, X. Wang, and X. Gan, "Critical sensing range for mobile heterogeneous camera sensor networks," in *Proc. IEEE INFOCOM*, 2014, pp. 970–978.
[29] H. Ma, M. Yang, D. Li, Y. Hong, and W. Chen, "Minimum camera barrier coverage in wireless camera sensor networks," in *Proc. IEEE INFOCOM*, 2012, pp. 217–225.
[30] Z. Yu, F. Yang, J. Teng, A. C. Champion, and D. Xuan, "Local face-view barrier coverage in camera sensor networks," in *Proc. IEEE INFOCOM*, 2015, pp. 684–692.
[31] Q. Zhang, S. He, and J. Chen, "Toward optimal orientation scheduling for full-view coverage in camera sensor networks," in *Proc. IEEE Global Commun. Conf.*, 2016, pp. 1–6.
[32] X. Liu, B. Yang, S. Zhao, and Y. Fan, "Achieving full-view barrier coverage with mobile camera sensors," in *Proc. Int. Conf. Netw. Netw. Appl.*, 2016, pp. 73–76.
[33] Y. Wu and X. Wang, "Achieving full view coverage with randomly-deployed heterogeneous camera sensors," in *Proc. IEEE 32nd Int. Conf. Distrib. Comput. Syst.*, 2012, pp. 556–565.
[34] R. Yang, X. Gao, F. Wu, and G. Chen, "Distributed algorithm for full-view barrier coverage with rotatable camera sensors," in *Proc. IEEE Global Commun. Conf.*, 2015, pp. 1–6.
[35] Y. C. Wang, Y. Chen, and Y. Tseng, "Using rotatable and directional (R & D) sensors to achieve temporal coverage of objects and its surveillance application," *IEEE Trans. Mobile Comput.*, vol. 11, no. 8, pp. 1358–1371, Aug. 2012.
[36] G. Fusco and H. Gupta, "Selection and orientation of directional sensors for coverage maximization," in *Proc. 6th Annu. IEEE Commun. Soc. Conf. Sensor Mesh Ad Hoc Commun. Netw.*, 2009, pp. 1–9.
[37] L. Guo *et al.*, "Enhancing barrier coverage with β quality of monitoring in wireless camera sensor networks," *Ad Hoc Netw.*, vol. 51, pp. 62–79, 2016.
[38] N. Joubert *et al.*, "Towards a drone cinematographer: Guiding quadrotor cameras using visual composition principles," *CoRR*, vol. abs/1610.01691, 2016.
[39] T. Nägeli *et al.*, "Real-time planning for automated multi-view drone cinematography," *ACM Trans. Graph.*, vol. 36, 2017, Art. no. 132.
[40] 2016. [Online]. Available: https://enterprise.dji.com/energy
[41] W. Xiaoyu, H. Dai, H. Huang, Y. Liu, G. Chen, and W. Dou, "Robust scheduling for wireless charger networks," in *Proc. IEEE INFOCOM*, 2019, pp. 2323–2331.
[42] D. Haipeng *et al.*, "Radiation constrained scheduling of wireless charging tasks," in *Proc. 18th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2017, Art. no. 17.
[43] S. Bhattacharya, A. Saifullah, C. Lu, and G. Roman, "Multi-application deployment in shared sensor networks based on quality of monitoring," in *Proc. 16th IEEE Real-Time Embedded Technol. Appl. Symp.*, 2010, pp. 259–268.
[44] I. F. Akyildiz *et al.*, "On exploiting spatial and temporal correlation in wireless sensor networks," in *Proc. Wiopt Model. Optim. Mobile Ad Hoc Wireless Netw.*, 2004, pp. 71–80.
[45] M. C. Vuran *et al.*, "Spatio-temporal correlation: Theory and applications for wireless sensor networks," *Elsevier Comput. Netw.*, vol. 45, pp. 245–259, 2004.
[46] T. Cover *et al.*, *Elements of Information Theory*. Hoboken, NJ, USA: Wiley, 1991.
[47] G. Carlos *et al.*, "Near-optimal sensor placements in Gaussian processes," in *Proc. 22nd ACM Int. Conf. Mach. Learn.*, 2005, pp. 265–272.
[48] A. Krause *et al.*, "Near-optimal sensor placements: Maximizing information while minimizing communication cost," in *Proc. 5th Int. Conf. Inf. Process. Sensor Netw.*, 2006, pp. 2–10.
[49] S. Tang and L. Yang, "Morello: A quality-of-monitoring oriented sensing scheduling protocol in sensor networks," in *Proc. IEEE INFOCOM*, 2012, pp. 2676–2680.
[50] S. Tang and L. Yang, "DAMson: On distributed sensing scheduling to achieve high quality of monitoring," in *Proc. IEEE INFOCOM*, 2013, pp. 155–159.
[51] S. Fujishige, *Submodular Functions and Optimization*. Amsterdam, The Netherlands: Elsevier, 2005.
[52] W. Wang, H. Dai, C. Dong, F. Xiao, X. Cheng, and G. Chen, "VISIT: Placement of unmanned aerial vehicles for anisotropic monitoring tasks," in *Proc. 16th Annu. IEEE Int. Conf. Sens. Commun. Netw.*, 2019, pp. 1–9.
[53] R. Tan *et al.*, "Exploiting data fusion to improve the coverage of wireless sensor networks," *IEEE/ACM Trans. Netw.*, vol. 20, no. 2, pp. 450–462, 2012.
[54] 2018. [Online]. Available: https://github.com/yestinsong/text-detection
[55] 2018. [Online]. Available: http://dlib.net/files/
[56] V. Chvatal, "A greedy heuristic for the set-covering problem," *Math. Operations Res.*, vol. 4, pp. 233–235, 1979.
[57] X. Wang *et al.*, "Practical heterogeneous wireless charger placement with obstacles," *IEEE Trans. Mobile Comput.*, vol. 19, no. 8, pp. 1910–1927, Aug. 2020.
[58] C. Wu, Y. Xu, Y. Chen, and C. Lu, "Submodular game for distributed application allocation in shared sensor networks," in *Proc. IEEE INFOCOM*, 2012, pp. 127–135.
[59] S. Tang *et al.*, "Qute: Quality-of-monitoring aware sensing and routing strategy in wireless sensor networks," in *Proc. 14th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2013, pp. 119–126.
[60] W. Wang *et al.*, "PANDA: Placement of unmanned aerial vehicles achieving 3D directional coverage," in *Proc. IEEE INFOCOM*, 2019, pp. 1198–1206.

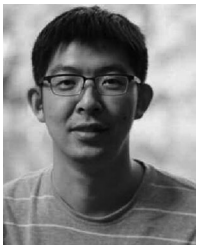
Weijun Wang (Student Member, IEEE) received the BS degree from the Department of Computer and Software, Nanjing University of Post and Telecommunication, Nanjing, China, in 2014, and the ME degree from the Department of Communication Engineering, PLA University of Science and Technology, Nanjing, China, in 2017. From 2017 to 2019, he was with Nanjing University and he is currently with the Faculty of Mathematics and Computer Science, George-August University Goettingen. His research interests include UAV monitoring problem, recognitive radio, and MAC protocols in UAV hoc networks.

Haipeng Dai (Member, IEEE) received the BS degree from the Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai, China, in 2010, and the PhD degree from the Department of Computer Science and Technology, Nanjing University, Nanjing, China, in 2014. His research interests mainly include the areas of wireless charging, mobile computing, and data mining. He is currently a research assistant professor with the Department of Computer Science and Technology, Nanjing University. His research papers have been published in many prestigious conferences and journals such as ACM MobiHoc, ACM VLDB, ACM SIGMETRICS, IEEE INFOCOM, IEEE ICDCS, IEEE ICNP, ICPP, IEEE IPSN, the *IEEE Journal on Selected Areas in Communications*, the *IEEE/ACM Transactions on Networking*, the *IEEE Transactions on Mobile Computing*, the *IEEE Transactions on Parallel and Distributed Systems*, the *IEEE Transactions on Services Computing* and IEEE TOSN. He is a member of the ACM. He serves/ed as poster chair of the IEEE ICNP'14, TPC member of the IEEE ICNP'14, IEEE ICC'14-17, IEEE ICCCN'15-17 and the IEEE Globecom'14-17. He received Best Paper Award from IEEE ICNP'15, and Best Paper Award Candidate from IEEE INFOCOM'17.



Chao Dong (Member, IEEE) received the PhD degree in communication engineering from the PLA University of Science and Technology, Nanjing, China, in 2007. From 2008 to 2011, he worked as a postdoc with the Department of Computer Science and Technology, Nanjing University, China. From 2011 to 2017, he was an associate professor with the Institute of Communications Engineering, PLA University of Science and Technology, Nanjing, China. He is currently a full professor with the College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, China. His current research interests include D2D communications, UAVs swarm networking, and anti-jamming network protocol. He is a member of the ACM and the IEICE.

Fu Xiao (Member, IEEE) received the PhD degree in computer science and technology from the Nanjing University of Science and Technology, Nanjing, China, in 2007. He is currently a professor and a PhD supervisor with the School of Computer, Nanjing University of Posts and Telecommunications, Nanjing. He has authored more than 30 papers in related international conferences and journals, including INFOCOM, ICC, IPCCC, the *IEEE Journal on Selected Areas in Communications*, the *IEEE/ACM Transactions on Networking*, the *IEEE Transactions on Mobile Computing*, the *ACM Transactions on Embedded Computing Systems*, and the *IEEE Transactions on Vehicular Technology*. His main research interest includes Internet of Things. He is a member of the IEEE Computer Society and the Association for Computing Machinery.



Jiaqi Zheng (Member, IEEE) is currently a research assistant professor with the Department of Computer Science and Technology, Nanjing University, China. His research interests include computer networking, particularly data center networks, SDN/NFV, machine learning system, and online optimization. He was a research assistant with the City University of Hong Kong, in 2015 and collaborated with Huawei Noah's Ark Lab. He visited CIS Center, Temple University, in 2016. He received the Best Paper Award from IEEE ICNP 2015, Doctorial Dissertation Award from ACM SIGCOMM China and Outstanding Doctorial Dissertation Award from CCF. He is a member of the ACM.

Xiao Cheng received the BS degree in mathematics and computing science from the Changsha University of Science and Technology, Changsha, China, in 2015. He is currently working toward the MS degree in computer science and technology at Naning University, Nanjing, China. His research interests include unmanned aerial vehicle communications and ad hoc networks.

Guihai Chen (Member, IEEE) received the BS degree in computer software from Nanjing University, Nanjing, China, in 1984, the ME degree in computer applications from Southeast University, Nanjing, China, in 1987, and the PhD degree in computer science from the University of Hong Kong, Hong Kong, in 1997. He is currently a professor and deputy chair with the Department of Computer Science, Nanjing University, China. He had been invited as a visiting professor by many foreign universities including the Kyushu Institute of Technology, Japan, in 1998, the University of Queensland, Australia, in 2000, and the Wayne State University, Detroit, Michigan during September 2001 to August 2003. He has a wide range of research interests with focus on sensor networks, peer-to-peer computing, high-performance computer architecture, and combinatorics.

Xiaoming Fu (Senior Member, IEEE) received the PhD degree in computer science from Tsinghua University, Beijing, China, in 2000. He is currently a professor and the head of the Computer Networks Group, University of Göttingen. He has also held visiting positions with ETSI, University of Cambridge, Columbia University, Tsinghua University, and UCLA. He is also a distinguished lecturer of the IEEE, a member of the ACM, a fellow of the IET, and a member of the Academia Europaea.