

Research Article

Two-Phase Algorithm for Optimal Camera Placement

Jun-Woo Ahn,¹ Tai-Woo Chang,² Sung-Hee Lee,³ and Yong Won Seo¹

¹College of Business & Economics, Chung-Ang University, 84 Heukseok-ro, Dongjak-gu, Seoul 06974, Republic of Korea

²Department of Industrial & Management Engineering, Kyonggi University, 154-42 Gwanggyosan-ro, Yeongtong-gu, Suwon, Gyeonggi 16227, Republic of Korea

³Department of Business Administration, Hoseo University, 12 Hoseodae-gil, Dongnam-gu, Cheonan-si, Chungcheongnam-do 31066, Republic of Korea

Correspondence should be addressed to Yong Won Seo; seoyw@cau.ac.kr

Received 17 May 2016; Accepted 19 July 2016

Academic Editor: Minsoo Kim

Copyright © 2016 Jun-Woo Ahn et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

As markers for visual sensor networks have become larger, interest in the optimal camera placement problem has continued to increase. The most featured solution for the optimal camera placement problem is based on binary integer programming (BIP). Due to the NP-hard characteristic of the optimal camera placement problem, however, it is difficult to find a solution for a complex, real-world problem using BIP. Many approximation algorithms have been developed to solve this problem. In this paper, a two-phase algorithm is proposed as an approximation algorithm based on BIP that can solve the optimal camera placement problem for a placement space larger than in current studies. This study solves the problem in three-dimensional space for a real-world structure.

1. Introduction

The global surveillance camera market is rapidly growing. According to the 2013 IMS Research data shown in Figure 1, the surveillance camera market is expected to grow by 1.5 times or more in the next five years. This is because surveillance cameras are used for more than simply preventing and solving crime or managing traffic. They are now needed for production assembly lines or observing natural disasters [1, 2]. Moreover, with the development in big data image-processing techniques, it is also possible not only to watch the images but also to extract the necessary data from them [3].

Along with the growth of the surveillance camera market, interest in efficient camera placement has also been increasing. If the placement of cameras is inefficient, even with many installed cameras the effect can be unsatisfactory. For efficient placement of surveillance cameras, several studies [4–15] have investigated the optimal camera placement problem. The optimal camera placement problem, sometimes called the camera network deployment problem, is defined as how to adequately place cameras to maximize the coverage under certain conditions [6, 10]. This optimal camera placement

problem consists of finding the minimum number of cameras that satisfies a specific coverage or finding the maximum coverage with a given number of cameras [4].

Current studies [6, 7, 10, 11] hypothesized a continuous space that is simplified as a two-dimensional (2D) grid of points. Here, the grid points are discrete points on x - and y -axes by the minimum distance Δ , which takes into account the spatial sampling frequency ($f_x = f_y : \Delta = 1/f_x = 1/f_y$) after simplifying real space into 2D [6]. When modeling a fixed-area terrain using the above method, the solution quality of the optimal camera placement problem with a higher resolution tends to be better than that with a lower resolution, because the ratio of the real-world terrain that is reflected in the modeling area with a high resolution (large f_a ; small Δ) using a larger number of grid points is higher than that with a low resolution (small f_a ; large Δ) using fewer grid points. Thus Hörster and Lienhart [6] claimed that considering a large number of grid points is necessary.

Because the optimal camera placement problem is NP-hard [16], existing studies have focused on finding efficient and effective approximation algorithms rather than finding an optimal solution.

TABLE 1: Publications on the optimal camera placement problem.

Source	Year	Methods	x grid	y grid	Number of grid points
[10] Surveillance of a 2D plane area with 3D deployed cameras	2014	PSO	4	4	16
[11] Particle swarm optimization inspired probability algorithm for optimal camera network placement	2012	PSO	11	11	121
[6] Approximating optimal visual sensor placement	2006	IP	12	12	144
[13] Grid coverage for surveillance and target location in distributed sensor networks	2002	IP	13	13	169
[7] Optimal camera placement for total coverage	2009	IP	12	6	72
Model simulation 1 in this paper		Greedy	200	200	40,000
Model simulation 2 (Appendix) in this paper		Greedy	500	500	250,000

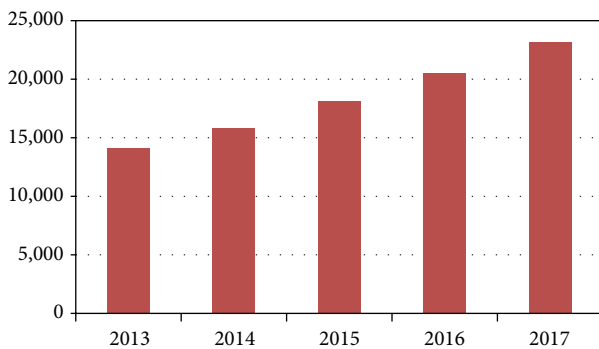


FIGURE 1: Surveillance Camera Market Size Prediction [20].

The approximation algorithm proposed in previous studies solves the problem directly at the high resolution of the desired level. On the other hand, our study proposes a method of finding a solution under a low resolution using BIP, then solving the problem correctly at its desired high resolution based on the found solution. The proposed method decreases the complexity of the calculation, which can lead to faster problem-solving at a high resolution than existing methods.

The reliability of setting the start point, which can cause a localized optimization in the approximate algorithm, is also improved. As a result, under the same conditions, the confidence of the proposed solution increases when compared to solving the problem at a high resolution to begin with.

Additionally, rather than using the virtual modeling area generally used in existing studies, this study uses a real-world modeling area from geographic information system (GIS) data of actual terrain. The data came from the satellite pictures. Three-dimensional (3D) camera placement was selected to provide more practicality, instead of 2D camera placement which is unrealistic to apply.

This paper is organized as follows. Section 2 analyzes the relevant studies. Section 3 explains the spatial configuration required for the camera placement and the calculation method for the surveillance camera view and also describes the algorithm that solves the actual problem. Section 4 compares the quality of the solutions obtained from binary integer

programming and from the proposed method. Section 5 presents the conclusion.

2. Literature Review

The art gallery problem (AGP), studied in the field of computational geometry, is the problem of placing at least one security guard to check every area of a museum or gallery. Because AGP finds the optimal placement point within the restricted viewpoint of the security guard and the optimal camera placement problem finds the optimal placement point within the restricted viewpoint of the camera, solving the optimal camera placement problem is very similar to solving AGP [17, 18].

This optimal camera placement problem has been studied to solve both MIN problem, which finds the minimum number of cameras and placement conditions to satisfy the target coverage under the given conditions, and the FIX problem, which maximizes the coverage with a fixed number of cameras under the given conditions [4].

In the meantime, looking from the methodological viewpoint of problem-solving, previous studies on solving the optimal camera placement problem generally have been based on binary integer programming (BIP) [5–9]. BIP offers the global optimal solution; however, the studies based on BIP only answer problems with limited, simple conditions due to the NP-hard property of the problem [4].

Therefore, studies have approached the problem from various directions to solve the optimal camera placement problem within a modeling area that can reflect reality with complex conditions, and many approximation algorithms have been suggested as a result [4–15]. Previous literature in the modeling area and the camera installation area has its roots in 2D-based studies [12]. The greedy algorithm [8, 14], genetic algorithm (GA) [10, 15], particle swarm optimization (PSO) [11, 12], and so on have been used in existing studies as approximation algorithms to solve the problem. However, all the studies mentioned above have high computational complexity, for they found the solution directly at a high resolution. Table 1 lists the approximation algorithms suggested in previous studies.

Moreover, the 2D model is too simple to compute a real-world case of the optimal camera placement problem [12]; methods to solve the problem using 3D were studied in

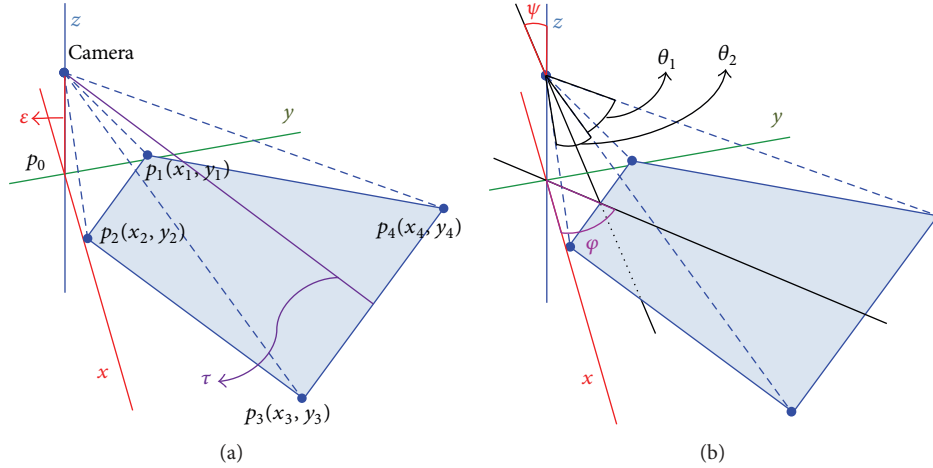


FIGURE 2: FOV computation parameters.

[11, 12]. However, 3D problem-solving exacerbates the issue of high computational complexity.

Previous studies have consistently reported the issue of high computational complexity as they continue to use problem-solving methods at high resolution. To remove this issue, phase 1 of the two-phase algorithm proposed in this study uses BIP to find the global optimal solution of the MIN problem within the low-resolution area (small number of grid points), and phase 2 uses an approximation algorithm, hill climbing method, to solve the FIX problem at a high resolution (large number of grid points).

With this process, the solution for a wider high-resolution area can be found based on the verified global optimal solution found in the low-resolution. Existing studies mainly used methods to avoid local optima, such as the genetic algorithm, particle swarm optimization, and simulated annealing, though they have high computational complexity [4]. Therefore, this study proposes using a hill climbing method, known to have low computational complexity. In general, greedy algorithms like a hill climbing method can find local optima if they are assigned the wrong starting point; however, this study proposes using the starting point found by BIP. The low computational complexity can reflect the modeling area of a large number of grid points with the same condition. Thus, this study proposes an approximation algorithm that is more likely to be used for real-world cases.

3. Model and Solution

This paper proposes a two-phase algorithm and assumes a 3D camera installation in a 2D modeling area. Phase 1 solves the problem using BIP, which offers an optimal solution by configuring the modeling area with a low-resolution grid for simple execution. Phase 2 finds a real-world applicable answer by setting the starting point from the low-resolution solution of phase 1 and then using the hill climbing method [19] for the modeling area configured with high-resolution grids.

3.1. Modeling Space. This paper assumes the surveillance of a plane area without obstacles. The surveillance area is divided into grid points, as shown in [13], and a grid point is captured by the camera if it is observed from the camera. As mentioned above, grid points refer to discrete points on x - and y -axes, separated by minimum distance Δ for the spatial sampling frequency [6]. Later, the plane area is divided into camera-installable and not camera-installable areas, and the surveillance area is assigned.

3.2. Modeling Surveillance Area. As in previous studies [4–12], field of view (FOV) modeling is proposed prior to explaining the placement method. Finding a solution for the optimal camera placement problem is equivalent to finding the conditions that create the FOV of each properly placed camera; the problem can be solved only if the method of computing the FOV is defined beforehand.

Like the study in [6], this study assumes a camera that is fixed in a certain direction so that it only surveils the same spot; therefore, a single camera has a fixed FOV depending on its installation condition. The FOV of the surveillance camera has a trapezoidal shape on the surveilled plane area, corresponding to the installation location (x_0, y_0) , horizontal angle (φ) , vertical angle (ψ) , installation height (ε) , horizontal and vertical angles of camera view (θ_1, θ_2) , and maximum recognition distance (T) . The horizontal and vertical angles of camera view mean the horizontal and vertical viewing angles of the scene captured by the camera.

Figure 2(a) shows the location of a camera which is installed at the ground coordinate $P_0(x_0, y_0)$ with the height ε and the recognition distance τ . Note that the actual recognition distance (τ) is less than or equal to the maximum recognition distance $(\tau \leq T)$. Figure 2(b) shows the horizontal view angle (θ_1) and the vertical view angle (θ_2) , as well as the horizontal angle (φ) and the vertical angle (ψ) . Here, the horizontal angle (φ) of the camera means the direction in which the camera watches. The vertical angle (ψ) is the watching angle of the camera, measured from a line perpendicular to the ground at the installation point.

TABLE 2: Camera specifications for the comparison test.

Consideration	Number of options	Details
Horizontal orientations (°)	8	0, 45, 90, ..., 270, 315
Vertical orientations (°)	15	1, 3, ..., 27, 29
Height (m)	1	7
Camera type		
Horizontal angle of view (°)	1	80
Vertical angle of view (°)	1	80
Available sight range (m)	1	60

Based on the given camera conditions, the algorithm to compute the coordinates of the trapezoid vertices, which are the FOV of the camera, is described as follows.

Step 0. $\tau = \varepsilon / \{\cos(\theta_2 + \psi)\}$. If $\theta_2 + \psi < 90$ or $\tau > T$, then stop the calculation.

Step 1. If $P_0 = (0, 0)$ and $\varphi = 0$, FOV is made of four vertices $\{P'_1, P'_2, P'_3, P'_4\}$ (each point P'_i is made of $\{x'_i, y'_i\}$):

$$\begin{bmatrix} h \times \tan \psi \\ \frac{h}{\cos \psi} \times \left\{ + \tan \left(\frac{\theta_1}{2} \right) \right\} \\ h \times \tan \psi \\ \frac{h}{\cos \psi} \times \left\{ - \tan \left(\frac{\theta_1}{2} \right) \right\} \\ h \times \tan (\psi + \theta_2) \\ \frac{h}{\cos (\psi + \theta_2)} \times \left\{ + \tan \left(\frac{\theta_1}{2} \right) \right\} \\ h \times \tan (\psi + \theta_2) \\ \frac{h}{\cos (\psi + \theta_2)} \times \left\{ - \tan \left(\frac{\theta_1}{2} \right) \right\} \end{bmatrix} = \begin{bmatrix} x'_1 \\ y'_1 \\ x'_2 \\ y'_2 \\ x'_3 \\ y'_3 \\ x'_4 \\ y'_4 \end{bmatrix}. \quad (1)$$

Step 2. Calculate vertex P''_i by rotating P'_i by φ

$$\begin{bmatrix} \cos \varphi & -\sin \varphi \\ \sin \varphi & \cos \varphi \end{bmatrix} \begin{bmatrix} x'_i \\ y'_i \end{bmatrix} = \begin{bmatrix} x''_i \\ y''_i \end{bmatrix}. \quad (2)$$

Step 3. Actual camera installation information is added to each P''_i

$$\begin{bmatrix} x''_i \\ y''_i \end{bmatrix} + \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} = \begin{bmatrix} x_i \\ y_i \end{bmatrix}. \quad (3)$$

Step 0 considers the maximum recognition distance (T), vertical angle (ψ), and vertical view angle (θ_2) to check whether the FOV can be computed. If τ exceeds the maximum recognition distance (T) set beforehand, FOV with such a condition does not exist and therefore is not computed. The FOV also does not exist if the sum of the vertical angle (ψ) and the vertical view angle (θ_2) exceeds 90 degrees, for the camera cannot see the floor.

Step 1 explains the calculation of the coordinates for the FOV trapezoid vertices, assuming that the surveillance camera is installed in a direction parallel to x -direction from the origin. Equation (1) takes the vertical angle (ψ) of the camera installation into account, as well as the vertical view angle (θ_2) and the horizontal view angle (θ_1).

Step 2 shows how to obtain the coordinates for the vertices of the FOV trapezoid by taking the horizontal angle (φ) of the installed camera into account, based on the value obtained from Step 1.

Step 3 includes the equation for calculating the coordinates for the vertices of the actual FOV trapezoid after adding the ground coordinates (x_0, y_0) of the installation point to the value from Step 2.

In conclusion, combining (1), (2), and (3) in Table 2 will consider the actual installation location for the camera and compute the coordinates of each vertex of the surveillance area (FOV trapezoid) of a single camera, using the matrix calculation of

$$\begin{bmatrix} \cos \varphi & -\sin \varphi & 0 & 0 & 0 & 0 & 0 & 0 \\ \sin \varphi & \cos \varphi & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \cos \varphi & -\sin \varphi & 0 & 0 & 0 & 0 \\ 0 & 0 & \sin \varphi & \cos \varphi & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \cos \varphi & -\sin \varphi & 0 & 0 \\ 0 & 0 & 0 & 0 & \sin \varphi & \cos \varphi & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cos \varphi & -\sin \varphi \\ 0 & 0 & 0 & 0 & 0 & 0 & \sin \varphi & \cos \varphi \end{bmatrix} \begin{bmatrix} h \times \tan \psi \\ \frac{h}{\cos \psi} \times \left\{ + \tan \left(\frac{\theta_1}{2} \right) \right\} \\ h \times \tan \psi \\ \frac{h}{\cos \psi} \times \left\{ - \tan \left(\frac{\theta_1}{2} \right) \right\} \\ h \times \tan (\psi + \theta_2) \\ \frac{h}{\cos (\psi + \theta_2)} \times \left\{ + \tan \left(\frac{\theta_1}{2} \right) \right\} \\ h \times \tan (\psi + \theta_2) \\ \frac{h}{\cos (\psi + \theta_2)} \times \left\{ - \tan \left(\frac{\theta_1}{2} \right) \right\} \end{bmatrix} + \begin{bmatrix} x_0 \\ y_0 \\ x_0 \\ y_0 \\ x_0 \\ y_0 \\ x_0 \\ y_0 \end{bmatrix} = \begin{bmatrix} x_1 \\ y_1 \\ x_2 \\ y_2 \\ x_3 \\ y_3 \\ x_4 \\ y_4 \end{bmatrix}. \quad (4)$$

3.3. *Two-Phase Algorithm.* Our two-phase algorithm (a) generates the grid model of the candidate locations of camera installation and the target area for surveillance; (b) solves the small scale of phase 1 at a low-resolution; (c) sets the starting value of phase 2 based on the solution from the previous step; and (d) solves the large scale problem of phase 2 at a high-resolution.

3.3.1. *Phase 1.* In phase 1, the minimum number of cameras that satisfies the specific condition given with the grid points of the simulation area is obtained; it is then used to solve the MIN problem to find the location of each camera and the installation condition. We approached the problem using a further-developed method based on BIP formulas [6] for the existing 2D placement problem to solve the 3D placement problem. The detailed procedure is as follows.

(1) First, decision variables are assigned, just as when solving a general BIP.

$$x_{ijdet} = \begin{cases} 1, & \text{If there exists a camera at position } i \text{ with horizontal orientation } j \text{ vertical orientation } d, \text{ height } e, \text{ and angle of view (AOV) } t, \\ 0, & \text{Otherwise,} \end{cases} \quad (5)$$

$$y_k = \begin{cases} 1, & \text{If the target at position } k \text{ is covered by a camera,} \\ 0, & \text{Otherwise.} \end{cases}$$

Thus, x_{ijdet} is 1 if there exists a camera at position i with horizontal angle j , vertical angle d , height e , and AOV t and 0 if not and y_k equals 1 if the surveillance area k is watched with x_{ijdet} and 0 if not.

(2) Other parameters required for the formula are defined as follows:

$$v_{ijdetk} = \begin{cases} 1, & \text{if the target at position } k \text{ is visible from camera position } i \text{ with horizontal orientation } j, \text{ vertical orientation } d, \text{ height } e, \text{ and AOV } t, \\ 0, & \text{Otherwise.} \end{cases} \quad (6)$$

N_C : number of camera positions.

N_{hD} : number of horizontal orientations.

N_{vD} : number of vertical orientations.

N_E : number of heights.

N_A : number of camera types.

N_T : number of target positions.

CVR: given minimal coverage rate.

(3) The objective function minimizes the number of cameras as follows:

$$\min \sum_{i=1}^{N_C} \sum_{j=1}^{N_{hD}} \sum_{d=1}^{N_{vD}} \sum_{e=1}^{N_E} \sum_{t=1}^{N_A} x_{ijdet}. \quad (7)$$

(4) The following constraints are also necessary:

$$\sum_{i=1}^{N_C} \sum_{j=1}^{N_{hD}} \sum_{d=1}^{N_{vD}} \sum_{e=1}^{N_E} \sum_{t=1}^{N_A} v_{ijdetk} \cdot x_{ijdet} \geq y_k \quad (8)$$

$$\text{for } \forall k(1, \dots, N_T),$$

$$\sum_{i=1}^{N_C} \sum_{j=1}^{N_{hD}} \sum_{d=1}^{N_{vD}} \sum_{e=1}^{N_E} \sum_{t=1}^{N_A} v_{ijdetk} \cdot x_{ijdet} \leq N_C \cdot y_k \quad (9)$$

for $\forall k(1, \dots, N_T)$,

$$\sum_{k=1}^{N_A} y_k \geq N_T \cdot \text{CVR}. \quad (10)$$

Equations (8) and (9) are the constraints to obtain y_k and (10) is a constraint such that the sum of y_k must be greater than or equal to the product of the minimum coverage rate and the target surveillance area.

3.3.2. *Phase 2.* Phase 2 solves the FIX problem, which finds the combination for maximum coverage with the constraint of the number of cameras determined from phase 1, with the result obtained from phase 1 as the initial value. This study uses the hill climbing method [19]. Since the objective function is nongradient, a direct-search method was applied; among the different direct-search methods, the alternating variable search method was used, as the problem has a multidimensional variable. This study proposes a hill climbing method, as [19] proposed such a method.

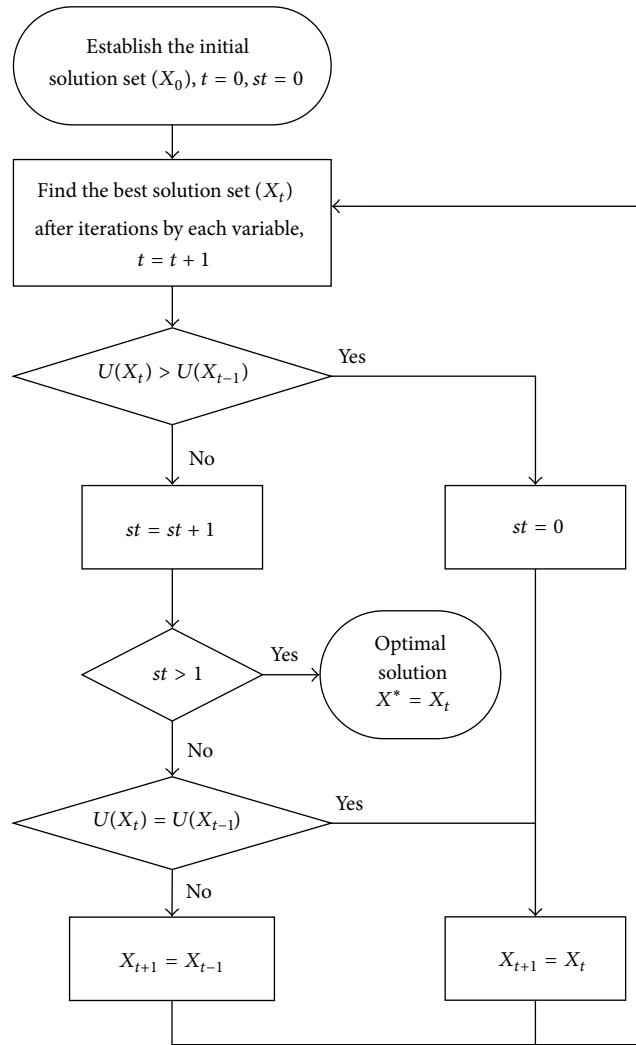


FIGURE 3: Flowchart of Phase 2.

Since phase 2 aims to maximize the coverage rate with the minimum number of cameras obtained in phase 1, the number of cameras does not change; only the conditions for each camera change. Each x_{ijdet} has the specification data for the x -coordinate, y -coordinate, horizontal angle (φ), vertical angle (ψ), installation height (ε), horizontal view angle (θ_1), and vertical view angle (θ_2) according to the camera type, as well as the maximum recognition distance. Each piece of information is both variable and dimensional. A flowchart of this phase is shown in Figure 3 and a hill climbing method of phase 2 is described as follows.

The notations and their meaning are as follows:

X : set of x_{ijdet}

X_t : set of optimal solutions found in the t th operation

t : the number of operations

st : the number of iterations that have an identical objective value

$U(X_t)$: coverage rate of camera set X at t th trial

n : index of a camera

m : index of a camera setting variable (one of i, j, d, e, t). (i has both x - and y -coordinate properties, therefore treated as two different variables).

Step 0 (initialization). Establish a starting point X_0^* and set $t = 0, st = 0$.

Step 1 (variant search). Set $t = t + 1$.

In X , the n th variable x_{ijdet} is selected. For each index (i, j, d, e, t) of x_{ijdet} , the variants are generated by numerically changing m th index of x_{ijdet} to the \pm direction. The objective function for each variant is evaluated, and the best one among the tried variants is taken upon comparing the values of the objective function.



(a) Picture of modeling area

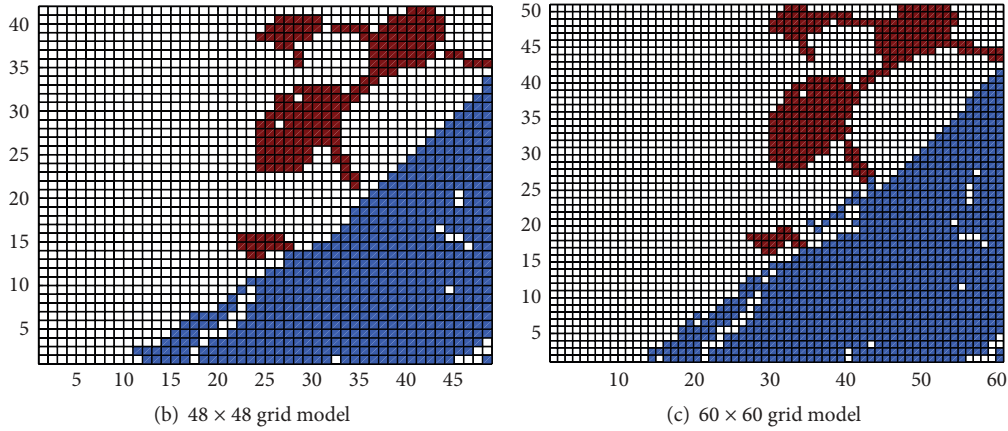


FIGURE 4: Map of Sevit Island (683 Olympic Blvd, Seocho-gu, Seoul, Korea) width 400 m; length 400 m.

Step 1.0 (initialization). Establish a starting point X_t and set $n = 0, m = 0$.

Step 1.1 (selection of a camera and calculation of the coverage rate)

For $n = 1: n(X_t)$;

Let $X_t^{\text{best}} = X_t$;

Let $x_n = n$ th variable in X_t

For $m = 1: 6$

Let X_{tmm+} to be a variant solution generated by numerically increase the value of the m th index of x_n , $U(X_{tmm+})$ to be the coverage rate of X_{tmm+} ;

Let X_{tmm-} to be a variant solution generated by numerically decrease the value of the m th index of x_n , and $U(X_{tmm-})$ to be the coverage rate of X_{tmm-} ;

Update X_t^{best} to be $\text{argmax}_{X_t \in \{X_t^{\text{best}}, X_{tmm+}, X_{tmm-}\}} (U(X))$, where in case of a tie one is randomly selected

End of For m

End of For n .

Step 1.2 (solution improvement). Set $X_t = X_t^{\text{best}}$.

Step 2 (check for improvement)

If $U(X_t) > U(X_{t-1})$, then set $X_{t+1} = X_t, st = 0$ and go to Step 1.

Otherwise set $st = st + 1$.

Step 3 (termination criterion)

If $st > 1$, then $X^* = X_t$; terminate the search.

Otherwise go to Step 4.

Step 4 (check for equivalent)

If $U(X_t) < U(X_{t-1})$, then set $X_{t+1} = X_{t-1}, st = 0$; and go to Step 1.

Otherwise set $X_{t+1} = X_t$ and go to Step 1.

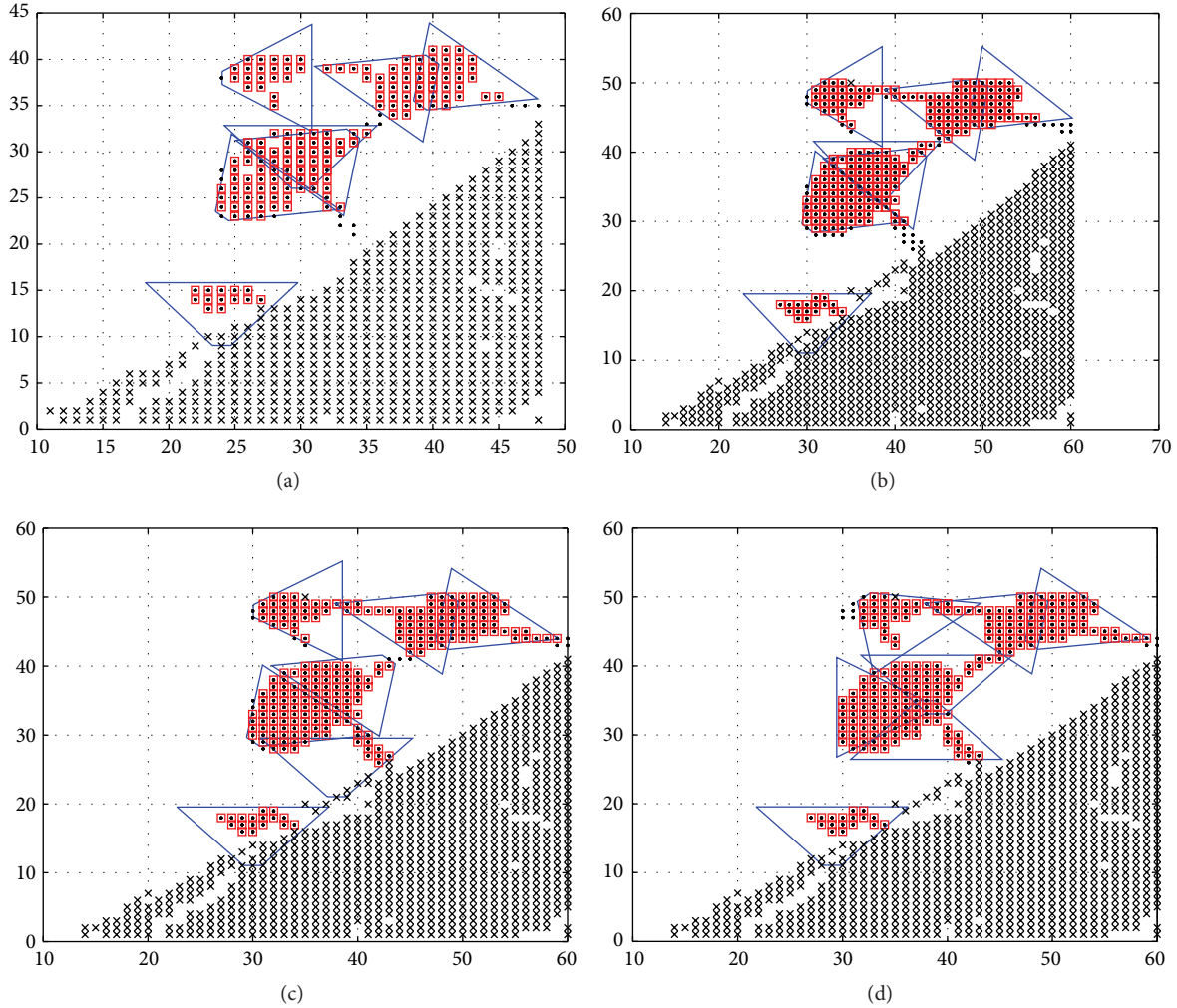


FIGURE 5: Solution for each step.

Step 0 sets the initial starting point. Here, t and st are set to 0.

Step 1 finds the best variant to improve the objective function, among the variants generated by increasing or decreasing the indices of each x_{ijdet} in the solution set X_t . Here, t is increased by 1. Step 1.0 initializes the current optimal solution X_t to be the starting point of this iteration. Step 1.1 finds the best solution which gives the highest coverage rate among the variants of X_t , generated by changing each of indices if x_{ijdet} is in X_t .

Step 2 checks if the coverage rate of X_t found in Step 1 actually increased compared to the coverage rate of X_{t-1} . If it is increased, X_{t+1} stays as X_t and st becomes zero. If not, st is increased by 1.

Step 3 checks whether the terminating conditions have been met. Here, if st is greater than 1, the optimal solution of phase 2 becomes $X^* = X_t$ and the solution search is completed.

Step 4 checks if the coverage rate of X_t decreased from the coverage rate of X_{t-1} . If decreased, X_{t+1} is reverted to X_{t-1} and moves to Step 1; if not, meaning that the values are the same, X_{t+1} stays at X_t and Step 1 is repeated.

The existing alternating variable search method [19] assigns the variable of the changing dimension; when it changes, the moving direction and the magnitude of change for the dimension variable where the objective function value improved the most are found. However, the method adapted for this study preassigns the magnitude of change for the dimension variable, and then finds which dimension variable in which direction improves the value of the objective function most. This method was utilized because the existing method vibrates near the optimal point, causing prolonged computation. Thus, it was more desirable to stop the computation at a proper point and use the solution than to continue the computation until it finished [19]. The method developed in this study eliminates the vibrating solution problem.

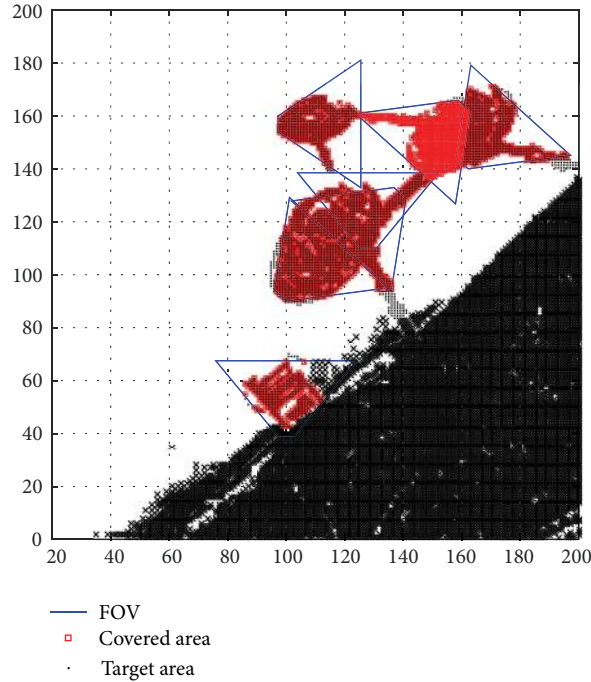


FIGURE 6: Solution obtained from high-resolution terrain.

4. Experimental Results

To evaluate the time efficiency and the coverage rate of the proposed two-phase algorithm, a comparison test was performed using a PC with an Intel Core i5-3337U processor, with 8 GB of DDR3 SDRAM. Matlab R2013b was used, and the BIP solution was obtained using the IBM ILOG CPLEX Optimization Studio 12.6.1. Additionally, the satellite map image of the actual landscape was transformed from an image to text (or numbers) with coordinates using Ascgen 2.0.0 from Jonathan Mathews Software.

To compare the global optimal solution obtained from BIP with a solution of the proposed two-phase algorithm, the modeling area was configured as in Figure 4. Figure 4(a) is the picture of the actual landscape of a $400\text{ m} \times 400\text{ m}$ square for the modeling area. Figure 4(b) is the modeling area transformed into 2304 low-resolution grids (48×48), and Figure 4(c) is the one transformed into 3600 high-resolution grids (60×60). It can be seen that the modeling area in Figure 4(c) with higher resolution reflects the actual landscape of Figure 4(a) better than the modeling area in Figure 4(b) at low-resolution. Since the geometric feature of the Sevit Island shown in Figure 4 is quite complicated requiring a high-resolution modeling, it is adequate to test the effectiveness of the proposed algorithm. Experimental results for three other problems are shown in the appendix.

The specifications of cameras for the comparison test are shown in Table 2. The horizontal angle for the camera l installation has eight options, starting from 0 degrees and stepping by 45 degrees; the vertical angle has 15 options, starting from 1 degree and stepping by 2 degrees. The height

of the installed camera is assumed to be 7 m, the horizontal view angle (θ_1) and vertical view angle (θ_2) are 80 degrees, and the maximum recognition distance is set to 60 m.

With the specifications shown in Table 2, a test that solves a real-world camera placement problem was carried out and the results were compared. As mentioned before, the two-phase algorithm consists of phase 1, which finds the initial solution (Figure 5(a)) using BIP with the specifications of Table 2 in the modeling area of Figure 4(b), and phase 2, which sets the starting point (Figure 5(b)) of the hill climbing method in the modeling area in Figure 4(c), based on the solution of Figure 5(b), and finds the solution (Figure 5(c)). Figure 5(d) is the solution directly obtained by BIP in the modeling area of Figure 4(c). Consequently, results from the proposed approximation algorithm and BIP can be evaluated, respectively, by comparing Figures 5(c) and 5(d).

The coverage rate of the solution and the time required for the test in case of Sevit Island area are as follows. When using the two-phase algorithm proposed in this study, the coverage rate is 94.72%, whereas it is 96.23% when using BIP. Therefore, comparing the solutions' coverage rates indicates that the proposed approximation algorithm obtains a solution with 98.43% of BIP's quality. The approximation algorithm took 15,823 ms, whereas BIP took 31,724 ms. Thus, the approximation algorithm needs about 49.88% of BIP's computing time.

As shown in these results, the coverage rate of the solution computed by the two-phase algorithm was comparable to that computed by BIP. This stems from the disadvantage of using the hill climbing method, which is simpler than the metaheuristic approximation used in previous studies, which

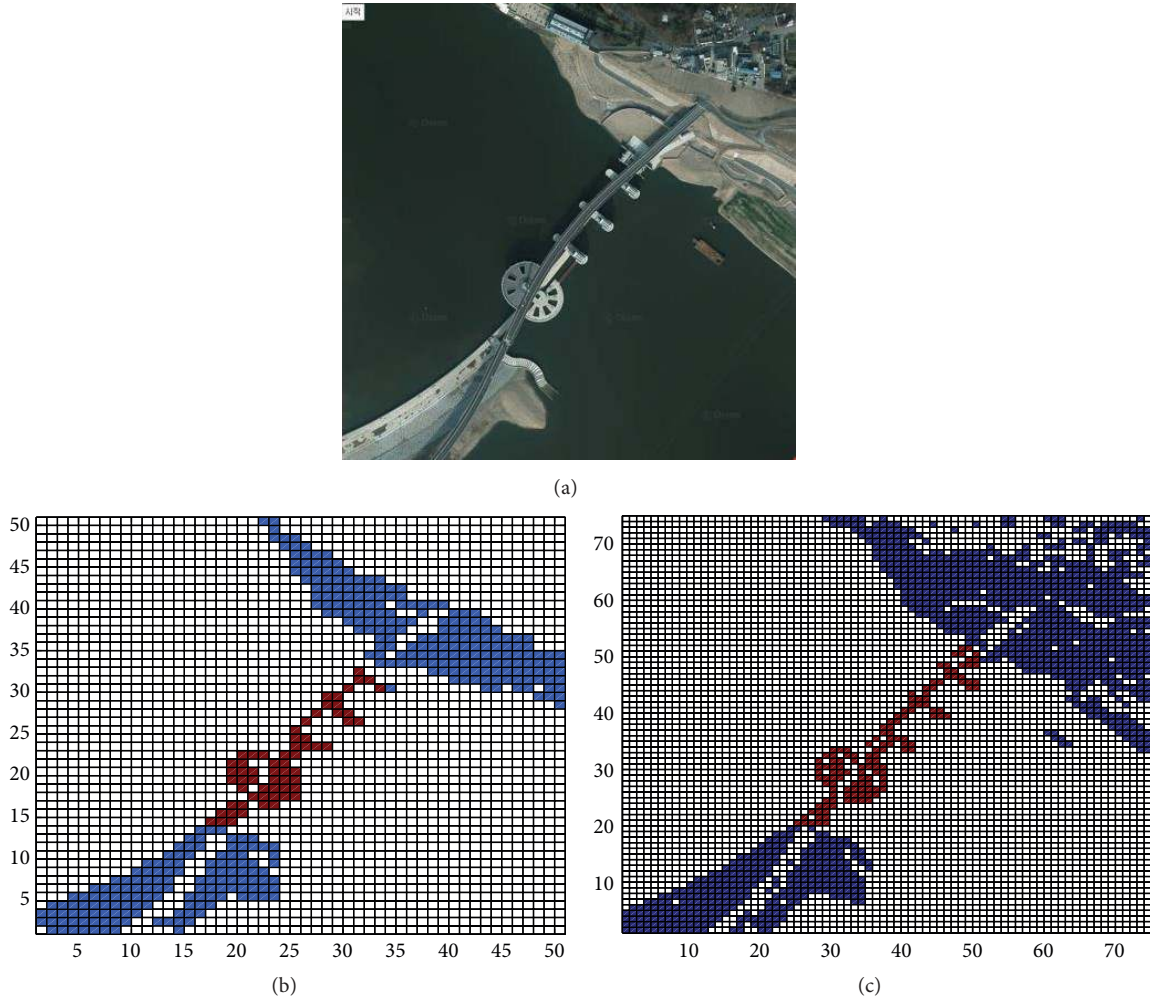


FIGURE 7: Map of Gangjeong Goryeong-bo (Daegu, Korea) 600 m \times 600 m square.

was complemented by using the optimal solution obtained by BIP in phase 1 as the starting point.

Moreover, the two-phase algorithm proposed in this study solved the problem more quickly than BIP. This means that the computational complexity of our proposed model is lower than that of BIP and, as mentioned in Section 2, our model is more adequate to a large-area problem, as well.

For the comparison study in this passage, phase 2 was performed with the area of a 3600 grid (60 \times 60), which is much smaller than the actual area that can be computed for a comparison of the solution quality. However, it would be more realistic and more accurate to approach with the higher-resolution area in phase 2, because the high-resolution terrain can reflect the actual landscape more precisely than the low-resolution terrain in the same space. Figure 6 shows the solution obtained by phase 2, performed in a higher-resolution terrain (40,000 grids (200 \times 200)) based on Figure 5(b). While existing studies have a difficulty in finding a solution for such a large terrain, the approximation algorithm proposed in this study can find a solution.

We performed the test not only with Sevit Island as a modeling area, but also with other modeling areas. Table 3 shows the comparison results of the two-phase algorithm (TPA) and BIP in the other modeling areas, of which solution details are described in Appendix.

This study was able to find the solution for terrains with a large number of grid points because it used phase 1 and phase 2. Phase 1 finds the global optimal solution using BIP at a low resolution, and phase 2 elaborates on the solution offered in phase 1 at a high resolution. This study's contribution is providing an effective method to solve the optimal camera placement problem for a wide, detailed area, which can be applied in real-world situations.

5. Conclusions

This study presented a two-phase approximation algorithm to solve the optimal camera placement problem. This algorithm had lower computational complexity than existing methods and did not reduce the quality of the solution. As a result,

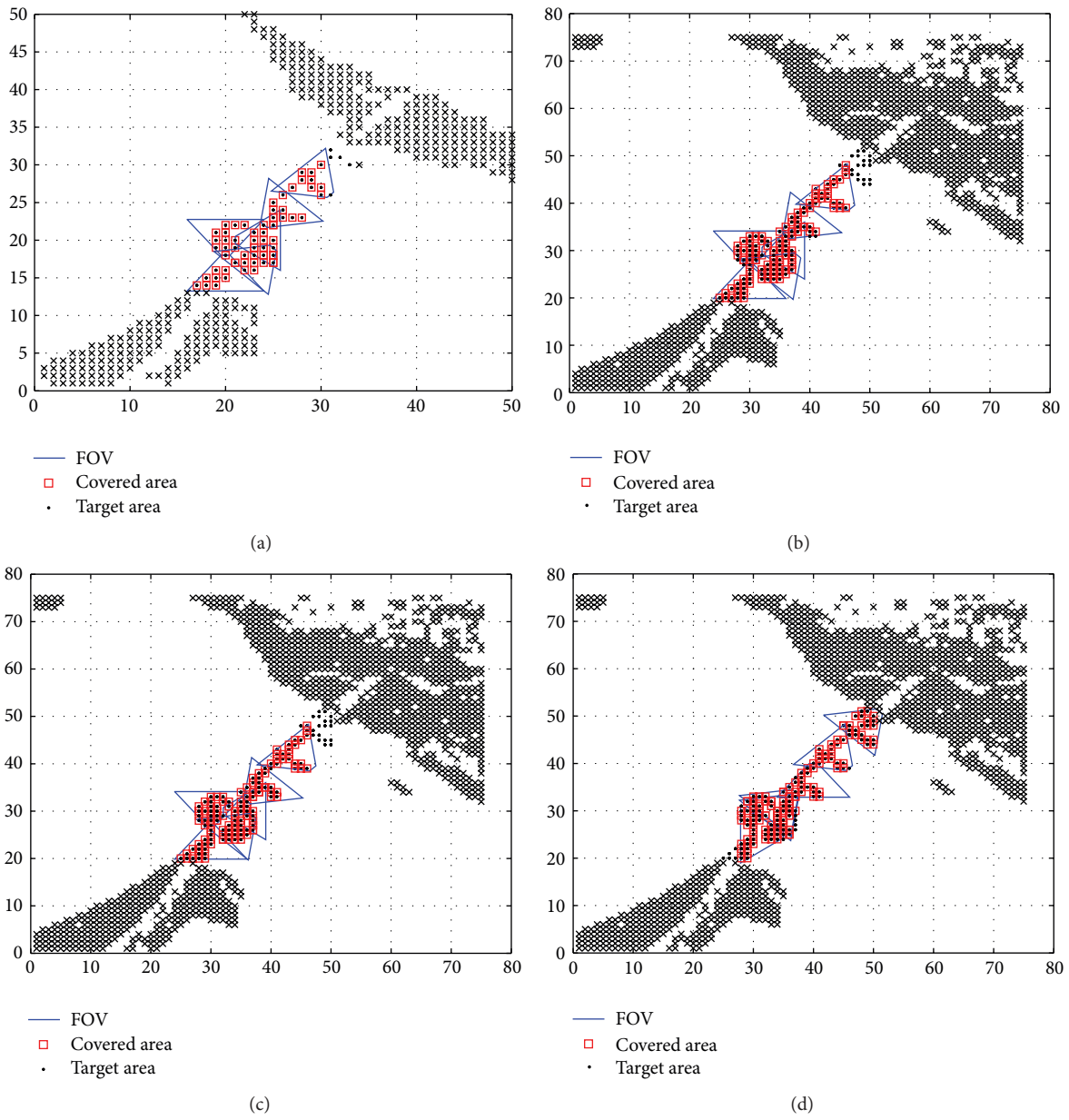


FIGURE 8: Solution to each step for Gangjeong Goryeong-bo (Weir) problem.

TABLE 3: Comparison test results.

Test area	Computing time (milliseconds)			Coverage rate (%)		
	TPA	BIP	TPA/BIP	TPA	BIP	TPA/BIP
Sevit Island	15,823	31,724	49.88%	94.72	96.23	98.43%
Gangjeong Goryeong-bo (weir)	4,378	17,030	25.71%	85.51	90.58	94.40%
Incheon port	9,896	32,665	30.30%	86.32	91.45	94.40%
Dongmyeong dock	31,403	48,785	64.37%	70.00	80.56	86.89%

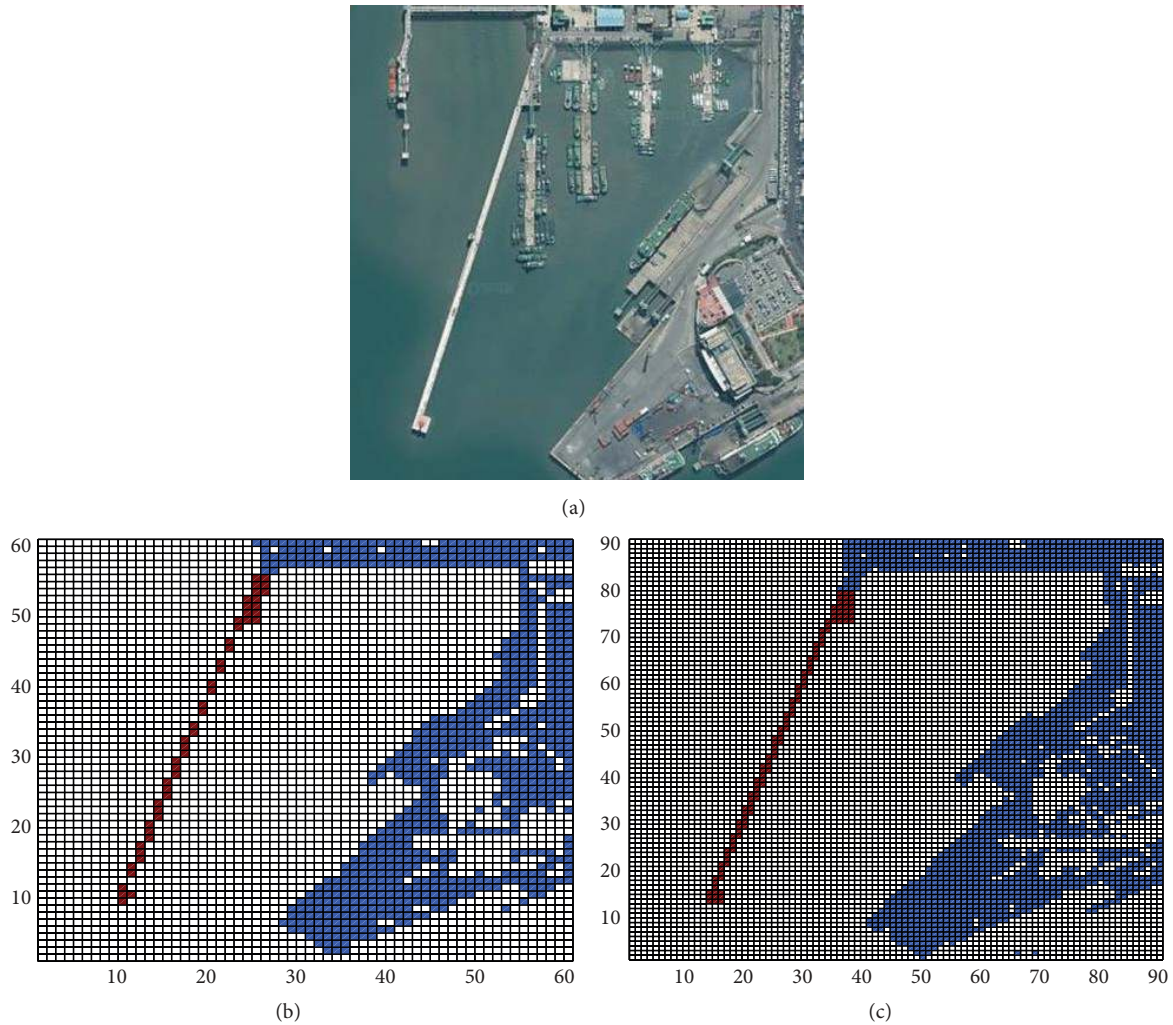


FIGURE 9: Map of Incheon International Airport Passenger Terminal (Incheon, Korea) $700\text{ m} \times 700\text{ m}$ square.

the optimal camera placement problem could be solved, even with wide, real-world terrain under complex conditions that could not be solved with the existing BIP method.

The two-phase algorithm proposed in this study finds a global optimal solution in phase 1 to use as the starting point in phase 2; thus, the confidence for the starting point is large. A comparison study in Section 4 reveals that the quality of the solution did not show significant differences from BIP.

Meanwhile, the two-phase algorithm proposed in this study had too high computational complexity to solve complex problems and thus could not reflect reality. However, the low-resolution problem could be solved using BIP, which not only offered a global optimal solution but also provided the idea of applying the solution of phase 1 in higher-resolution terrains that look more like reality. Phase 2 changed the method of solving the problem so that other perspectives could be applied to problems in the future. This study used a hill climbing method in phase 2 with low computational

complexity, but other methods, such as GA or PSO, could also be used in later studies.

The limitation of the study was that it used a hill climbing method in phase 2, which converges to relative local optimal solutions, instead of other approximate algorithms such as SA, GA, or PSO that have a higher chance of avoiding local optima. In this regard other approximate algorithms could also be applied in later studies. To solve the optimal camera placement problem with this modeling area with a large number of grid points under realistic restrictions, one of two approaches should be chosen: (i) to solve the problem through a simple algorithm at high resolution or (ii) to solve the problem at a relatively low resolution using a different method that needs more computational resources but has a higher change of finding a global optimal solution. This paper proposed the former to solve the problem at a high resolution in a wider terrain. Finding a proper balance point by comparing the two methods is left for future work.

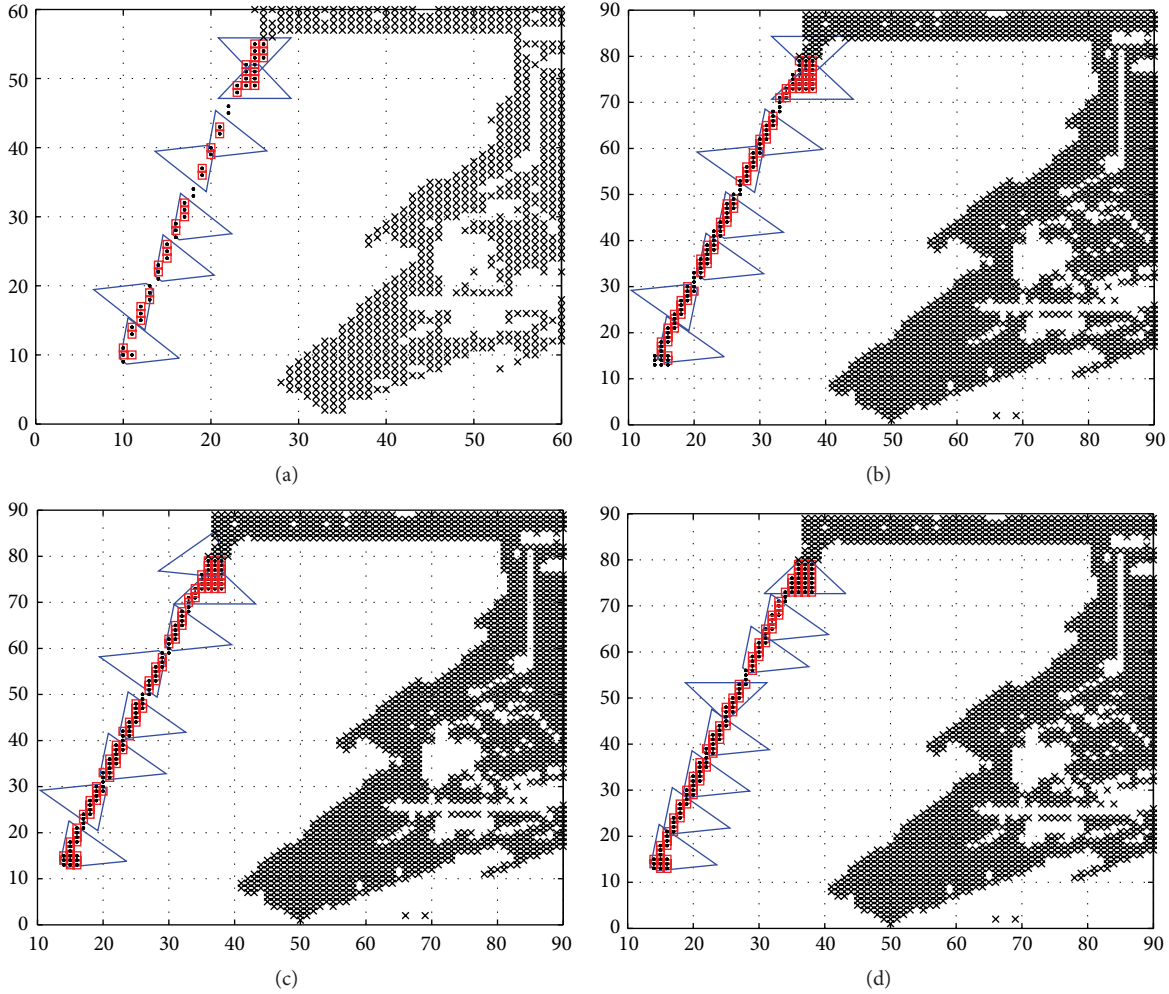


FIGURE 10: Solution to each step for the Incheon port problem.

Appendix

A. Additional Comparison Studies

The conditions for the additional comparison studies mentioned in Table 2 are identical to those described in Table 2.

A.1. Gangjeong Goryeong-bo (Weir) Problem. The modeling area in Figure 7 was configured to perform the model on Gangjeong Goryeong-bo (Weir) mentioned in Table 3. Figure 7(a) is the actual landscape of a 600 m × 600 m square to configure the modeling area. Figure 7(b) makes the 600 m × 600 m square terrain into a low-resolution modeling area of 2500 grids (50 × 50), and Figure 7(c) makes the same terrain into a high-resolution modeling area of 5625 grids (75 × 75).

A comparison test to solve the real-world optimal camera placement problem was carried out with the conditions mentioned above. Figure 8(a) shows the solution found in the modeling area in Figure 7(b), using BIP with the conditions of Table 2. Based on the solution of Figure 8(a), Figure 8(b) is the starting point of phase 2 for the hill climbing method in

the modeling area of Figure 7(c). Figure 8(c) is the solution obtained by performing phase 2. The solution obtained by performing BIP in the modeling area of Figure 7(c) to begin with is Figure 8(d). The difference between the problems solved using the approximation algorithm proposed in this study or BIP can be studied by comparing Figures 8(c) and 8(d). The computing times and coverage rates of the result can be confirmed in Table 3.

A.2. Incheon Port Problem. The modeling area in Figure 9 was configured for the model on Incheon port mentioned in Table 3. Figure 9(a) is the actual landscape of a 700 m × 700 m square to configure the modeling area. Figure 9(b) shows the 700 m × 700 m square terrain as a low-resolution modeling area of 3600 grids (60 × 60), and Figure 9(c) shows the same terrain as a high-resolution modeling area of 6400 grids (80 × 80).

A comparison test to solve the real-world optimal camera placement problem was carried out with the conditions mentioned above. Figure 10(a) shows the solution found in the modeling area in Figure 9(b) using BIP with the conditions

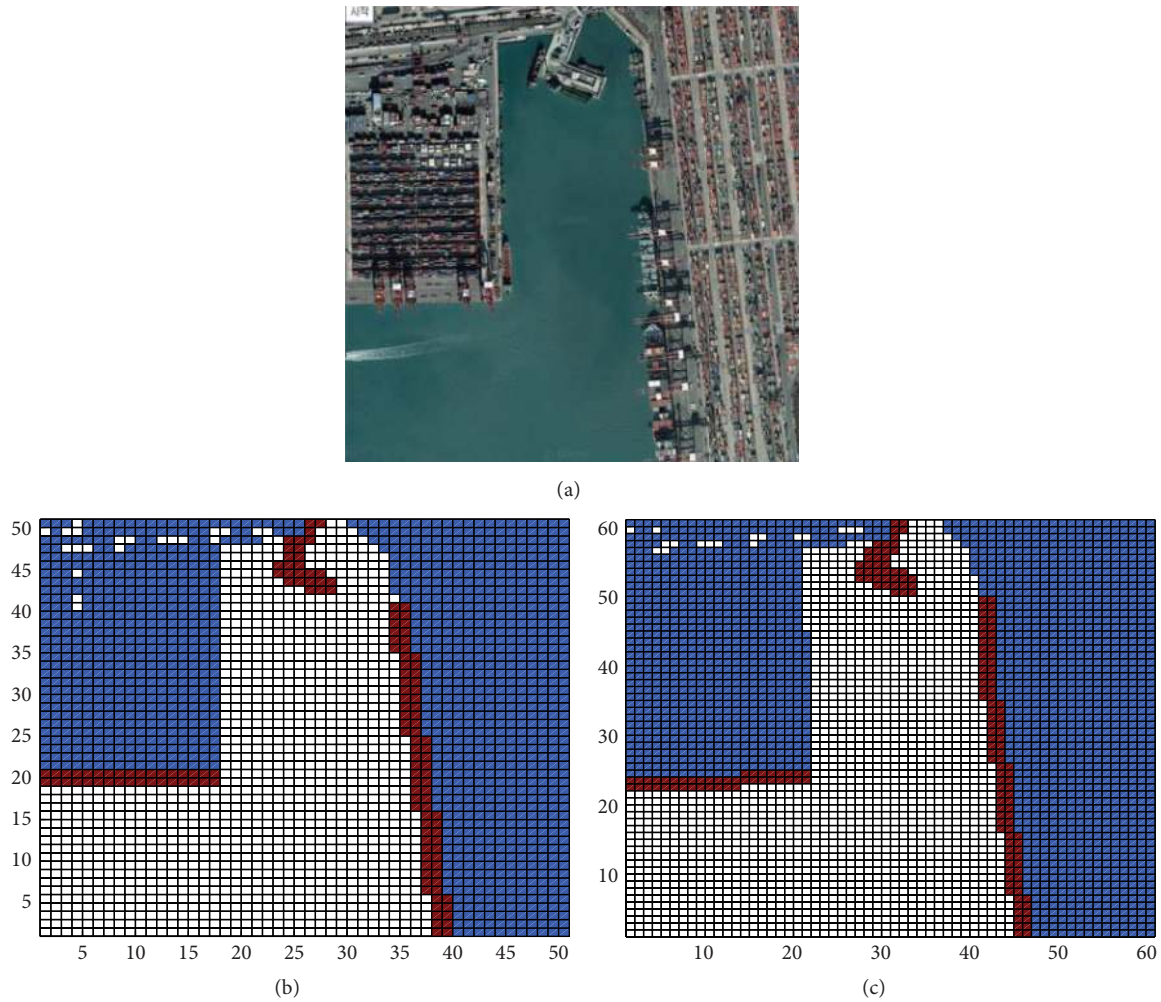


FIGURE 11: Map of Dongmyeong dock (Busan, Korea) 1000 m \times 1000 m square.

of Table 2. Based on the solution of Figure 10(a), Figure 10(b) is the starting point of phase 2 for the hill climbing method in the modeling area of Figure 9(c). Figure 10(c) is the solution obtained by performing phase 2. The solution obtained by performing BIP in the modeling area of Figure 9(c) to begin with is Figure 10(d). The difference between the problems solved using the approximation algorithm proposed in this study or BIP can be studied by comparing Figures 10(c) and 10(d). The computing times and coverage rates of the result can be confirmed in Table 3.

A.3. Busan Dongmyeong Dock Problem. The modeling area in Figure 11 was configured to perform the model on the Dongmyeong dock mentioned in Table 3. Figure 11(a) is the actual landscape of a 1000 m \times 1000 m square to configure the modeling area. Figure 11(b) shows the 1000 m \times 1000 m square terrain as a low-resolution modeling area of 2500 grids (50 \times 50), and Figure 11(c) shows the same terrain as a high-resolution modeling area of 3600 grids (60 \times 60).

A comparison test to solve the real-world optimal camera placement problem was carried out with the conditions mentioned above. Figure 12(a) shows the solution found in the modeling area in Figure 11(b) using BIP with the conditions of Table 2. Based on the solution of Figure 12(a), Figure 12(b) is the starting point of phase 2 for the hill climbing method in the modeling area of Figure 11(c). Figure 12(c) is the solution obtained by performing phase 2. The solution obtained by performing BIP in the modeling area of Figure 11(c) to begin with is Figure 12(d). The difference between the problems solved using the approximation algorithm proposed in this study or BIP can be studied by comparing Figures 12(c) and 12(d). The computing times and coverage rates of the result can be confirmed in Table 3.

Competing Interests

The authors declare that there are no competing interests regarding the publication of this paper.

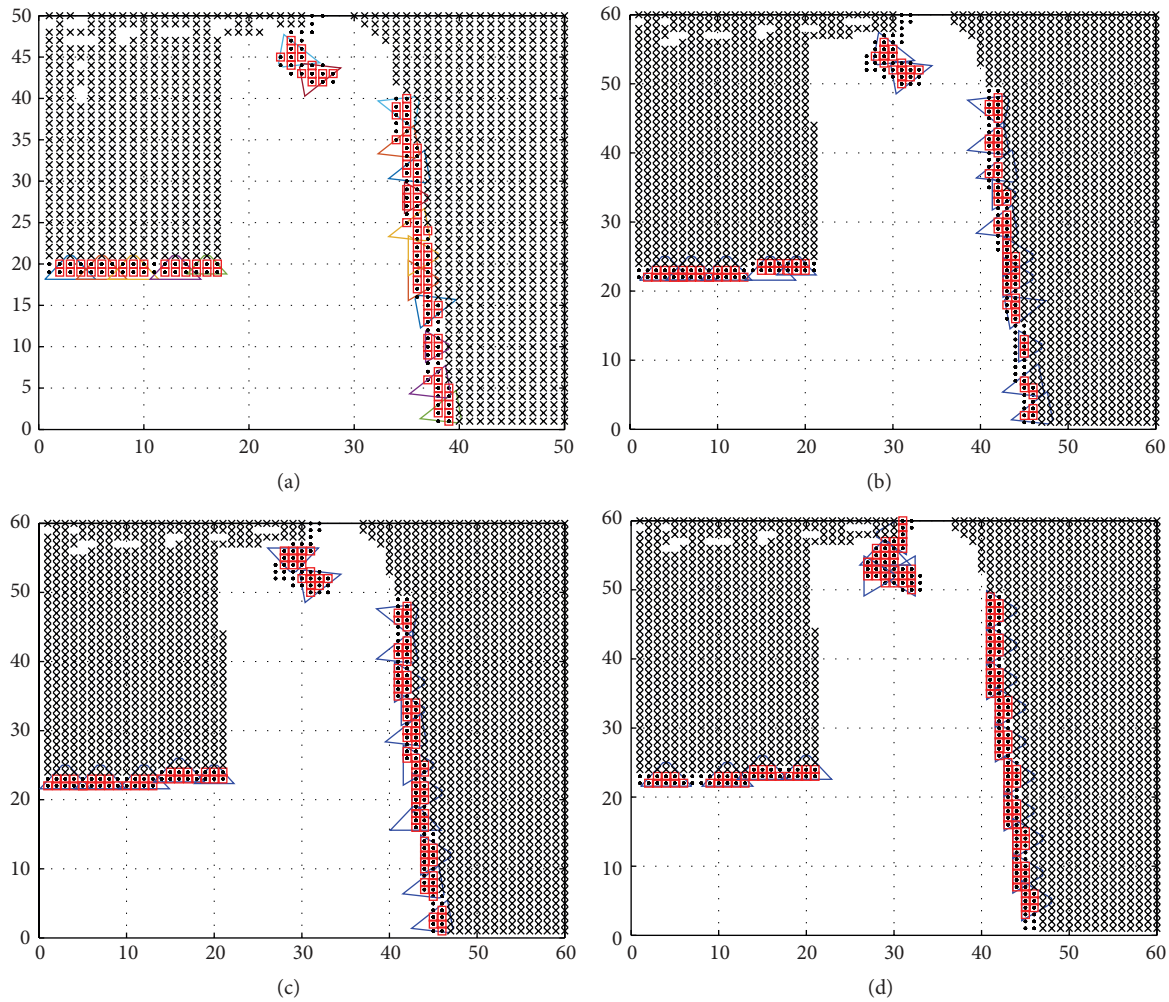


FIGURE 12: Solution to each step for the Busan Dongmyeong dock problem.

Acknowledgments

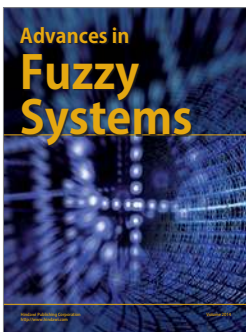
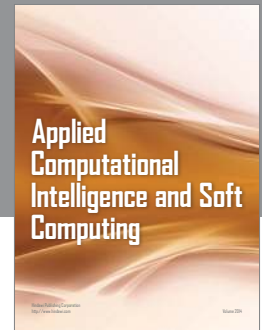
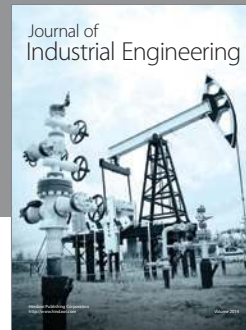
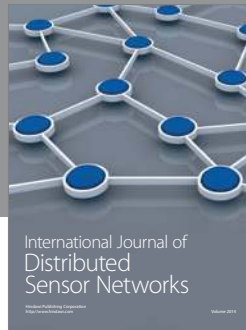
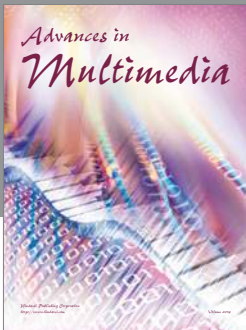
This research was supported by a grant (14SCIP-B065985-02) from Smart Civil Infrastructure Research Program funded by Ministry of Land, Infrastructure, and Transport (MOLIT) of Korea Government and Korea Agency for Infrastructure Technology Advancement (KAIA).

References

- [1] K. Lee, K. Yim, and M. A. Mikki, "A secure framework of the surveillance video network integrating heterogeneous video formats and protocols," *Computers and Mathematics with Applications*, vol. 63, no. 2, pp. 525–535, 2012.
- [2] N. Razmjooy, B. S. Mousavi, and F. Soleymani, "A real-time mathematical computer method for potato inspection using machine vision," *Computers and Mathematics with Applications*, vol. 63, no. 1, pp. 268–279, 2012.
- [3] D. I. D. Staff, *DARPA's VIRAT: Video Search, with a Twist*, Defense Industry Daily, 2010.
- [4] J. Zhao, R. Yoshida, S.-C. S. Cheung, and D. Haws, "Approximate techniques in solving optimal camera placement problems," *International Journal of Distributed Sensor Networks*, vol. 9, no. 11, Article ID 241913, 2013.
- [5] J. Ai and A. A. Abouzeid, "Coverage by directional sensors in randomly deployed wireless sensor networks," *Journal of Combinatorial Optimization*, vol. 11, no. 1, pp. 21–41, 2006.
- [6] E. Hörster and R. Lienhart, "Approximating optimal visual sensor placement," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME '06)*, pp. 1257–1260, July 2006.
- [7] J.-J. Gonzalez-Barbosa, T. García-Ramírez, J. Salas, J.-B. Hurtado-Ramos, and J.-D. Rico-Jiménez, "Optimal camera placement for total coverage," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '09)*, pp. 844–848, IEEE Press, Kobe, Japan, May 2009.
- [8] J. Zhao, S.-C. S. Cheung, and T. Nguyen, "Optimal visual sensor network configuration," in *Multi-Camera Networks: Principles and Applications*, H. Aghajan and A. Cavallaro, Eds., pp. 139–162, Academic Press, New York, NY, USA, 2009.
- [9] Z. Fang and J. Wang, "Hybrid approximation for minimum-cost target coverage in wireless sensor networks," *Optimization Letters*, vol. 4, no. 3, pp. 371–381, 2010.
- [10] A. Van Den Hengel, R. Hill, B. Ward et al., "Automatic camera placement for large scale surveillance networks," in *Proceedings*

of the Workshop on Applications of Computer Vision (WACV '09), pp. 1–6, Snowbird, Utah, USA, December 2009.

- [11] Y. Morsly, N. Aouf, M. S. Djouadi, and M. Richardson, “Particle swarm optimization inspired probability algorithm for optimal camera network placement,” *IEEE Sensors Journal*, vol. 12, no. 5, pp. 1402–1412, 2012.
- [12] Y.-G. Fu, J. Zhou, and L. Deng, “Surveillance of a 2D plane area with 3D deployed cameras,” *Sensors*, vol. 14, no. 2, pp. 1988–2011, 2014.
- [13] K. Chakrabarty, S. S. Iyengar, H. Qi, and E. Cho, “Grid coverage for surveillance and target location in distributed sensor networks,” *IEEE Transactions on Computers*, vol. 51, no. 12, pp. 1448–1453, 2002.
- [14] M. Al Hasan, K. K. Ramachandran, and J. E. Mitchell, “Optimal placement of stereo sensors,” *Optimization Letters*, vol. 2, no. 1, pp. 99–111, 2008.
- [15] X. Chen and J. Davis, “Camera placement considering occlusion for robust motion capture,” Tech. Rep. 2, Computer Graphics Laboratory, Stanford University, Stanford, Calif, USA, 2000.
- [16] R. Cole and M. Sharir, “Visibility problems for polyhedral terrains,” *Journal of Symbolic Computation*, vol. 7, no. 1, pp. 11–30, 1989.
- [17] J. O’Rourke, *Art Gallery Theorems and Algorithms*, Oxford University Press, Oxford, UK, 1987.
- [18] H. González-Baños, “A randomized art-gallery algorithm for sensor placement,” in *Proceedings of the 17th Annual Symposium on Computational Geometry (SCG '01)*, pp. 232–240, Medford, Massachusetts, USA, June 2001.
- [19] H.-P. P. Schwefel, *Evolution and Optimum Seeking: The Sixth Generation*, John Wiley & Sons, New York, NY, USA, 1993.
- [20] IHS Technology, Video Surveillance & Storage—Opportunities at the Intersection of IT & Physical Security, 2014.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

