Optimum Camera Placement
Considering Camera Specification
for Security Monitoring

Tokyo University of Agriculture and Technology

Kenichi Yabuta* and Hitoshi Kitazawa

* e-mail: kyabuta@m.ieice.org
In order to cover all the area of interest by a minimum number of cameras, we propose a new algorithm for optimum camera placement. This algorithm automate solving the camera placement considering camera specification: camera position, camera direction, and visual angle.
Scene Segmentation into Regions

1) Inputting top view scene data
2) Segmenting regions by extending block lines
3) Merging unnecessary divided regions to reduce computational time
4) Numbering all regions
Visibility Test

The visibility is decided by the line segment connecting region centers.

This line crosses no block. ⇒ There are visibility
The region A is visible from region B
The region B is visible from region A

This line crosses a block. ⇒ There are no visibility
The region A is invisible from region C
The region C is invisible from region A
The observed regions from camera candidate are decided.

**camera candidate**: The all combination of location, direction, and visual angle

The three regions can be observed from placed camera candidate.
The visibility between camera candidates and regions

Camera Placement

The visibility between camera candidates and regions

- **Camera candidate a** can observe region 1, 2, 4, ...
- **Camera candidate b** can observe region 2, 4, 5, ...
- **Camera candidate c** can observe region 1, 3, 5, ...

Set cover problem

<table>
<thead>
<tr>
<th>region</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>O</td>
<td>×</td>
<td></td>
<td>O</td>
</tr>
<tr>
<td>2</td>
<td>O</td>
<td>O</td>
<td>×</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>×</td>
<td>×</td>
<td>O</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>O</td>
<td>O</td>
<td>×</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>×</td>
<td>O</td>
<td>O</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Minimizing the number of camera on condition that all regions should be observed

Minimize

\[ Z = \sum_{i=1}^{n} x_i \]

Subject To:

\[ \sum_{i=1}^{n} p_{ij} x_i \geq 1 \quad \forall j \]

\[ x_i \in \{0,1\} \quad \forall i \]
Notation for stating the problem formally

\[ i = \text{index of camera candidates, \ } i = 1, 2, \ldots, n \]
\[ j = \text{index of observed regions, \ } j = 1, 2, \ldots, m \]

\[ x_i = \begin{cases} 
1 & \text{if camera candidate } i \text{ is placed.} \\
0 & \text{otherwise.}
\end{cases} \]

\[ p_{ij} = \begin{cases} 
1 & \text{if camera candidate } i \text{ can observe region } j. \\
0 & \text{otherwise.}
\end{cases} \]

\[ s_j = \begin{cases} 
1 & \text{if region } j \text{ is essential region.} \\
0 & \text{otherwise.}
\end{cases} \]

\[ y_j = \begin{cases} 
1 & \text{if region } j \text{ is not essential region and region } j \text{ is observed.} \\
0 & \text{otherwise.}
\end{cases} \]
ARO algorithm requires a large number of cameras.

In order to trace the flow of person, it is sufficient to observe cross-region. This region is called essential region.

Minimizing the number of camera on condition that all essential regions should be observed.
Minimize

\[ Z = \sum_{i=1}^{n} x_i - w_c \sum_{j=1}^{m} (1 - s_j)y_j \]

Subject To:

\[ \sum_{i=1}^{n} p_{ij} x_i \geq 1 \quad \text{if } s_j = 1 \]

\[ -y_j + \sum_{i=1}^{n} p_{ij} x_i \geq 0 \quad \text{if } s_j = 0 \]

\[ x_i \in \{0, 1\} \quad \forall i \]

\[ y_j \in \{0, 1\} \quad \forall j \]

To minimize the number of placed camera and maximize the number of observed regions.

Essential region should be observed and the \( y_j \) becomes 1 when other region is observed.
## Experimental Results

- The number of observed regions and placed camera, and processing time -

<table>
<thead>
<tr>
<th>Camera type</th>
<th>Camera candidate</th>
<th>Observed regions</th>
<th>Placed cameras</th>
<th>Processing time[sec]</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARO single</td>
<td>7704</td>
<td>107</td>
<td>22</td>
<td>0.500</td>
</tr>
<tr>
<td>ARO multiple</td>
<td>38520</td>
<td>107</td>
<td>20</td>
<td>1.063</td>
</tr>
<tr>
<td>WRO single</td>
<td>7704</td>
<td>82</td>
<td>11</td>
<td>0.328</td>
</tr>
<tr>
<td>WRO multiple</td>
<td>38520</td>
<td>85</td>
<td>10</td>
<td>0.844</td>
</tr>
</tbody>
</table>

The **WRO** algorithm can decrease the number of cameras.

The **WRO** algorithm can decrease the processing time.
-Correlation of processing time with the number of regions-

![Graph showing the correlation between processing time and the number of regions for ARO-single, ARO-multiple, WRO-single, and WRO-multiple categories.](image-url)
The WRO algorithm can reduce the number of placed cameras by 60% comparing ARO algorithm.
We have presented a novel algorithm for an optimum camera placement considering camera specification

- Our algorithm automatically decides camera position, direction, and visual angle
- To reduce the number of cameras weighting observed regions allocating multiple types of cameras
- Future works
  Considering network and power supply connections