Research on the Flight Path of the Drones Based on the 0-1 Linear Programming Model for the Light Show

Ying-ke CHEN, Yi-cheng WANG*, Xing-yue CHEN and Jia-xiu HAN
Hangzhou Foreign Languages School, Hangzhou, Zhejiang, China, 310023
*Corresponding author

Keywords: Drones, Flight path planning, Linear programming, Light show.

Abstract. With the rapid development of drone technology, using clusters of drones for aerial light shows has been widely welcomed in recent years. In this paper, several models were proposed to provide a wonderful light show. Firstly, this paper defined an operator template to discretize the images and get the location of drones in each images. Then, this paper calculated the distance matrix between images and proposed the optimal flight path of the drones to minimize the maximum distance between the adjacent images. Finally, this paper discussed the safe flight path of drones to avoid collisions.

Introduction

In 2015, 100 drones were launched simultaneously in Hamburg, Germany, creating a gorgeous pattern in the air. A year later, Intel developed its Shooting Star drone and used a cluster these drones of 500, controlled by a single laptop and one pilot, to perform a beautifully choreographed light show, which set a new record [1]. With this kind of Intel’s drones and software, it only takes a few days for users to create a light show. Moreover, Intel algorithm can automatically deal with animation production process and plan the quickest path to create an aerial pattern. With just one picture, it can quickly calculate the number of drones required and determine where the drones will be placed initially.

With the rapid development of drone technology, using clusters of drones for aerial light shows has been widely welcomed in recent years. Meanwhile, a satisfactory show effect requires a strong cluster control ground station and the coordinates of the cm level positioning, and while using GPS. It is possible to employ the orientation of multiple sets of radio, radar, light and shadow image positioning feedback and the corresponding analysis software [2]. And in addition to the guarantee of these devices, the sense of experience, the safety in the air, the degree of image forming and other similar consideration all need to be taken into account in the performance of the drone cluster as sky light show.

Discrete Model of Images

In order to show the position of each drone, the first thing is to rasterize the images which will be displayed on the sky. Each image can be discrete though MATLAB Image Viewer Toolbox and represented by the pixel matrix. Firstly, we import these images from MATLAB software. As a result, pixel matrix \((g = (g_{ij})_{m \times m})\) of each image will be obtained, which showed the image with number from 0 to 255, while “0” refers white, and “255” refers black. And, \(m \times m\) is the size of pixel matrix. For example, Fig.1 shows the initial image which will be displayed on the sky and Fig.2 shows the discrete result of the Fig.1 from MATLAB Image Viewer Toolbox.
Importing the Fig.1 from MATLAB [3], we will get a 215 × 215 pixel matrix \( G = (g_{ij})_{215 \times 215} \). Each element ranges from 0 to 255 to record the gray value of the image. In fact, we just need to identify each element whether is not white. To simplify the expression of the pixel matrix, this paper used a 0-1 matrix \( g_1 = (g_{1ij})_{215 \times 215} \) to represent. The relationship between two matrices as follows:

\[
g_{1ij} = \begin{cases} 
1 & \text{if } g_{ij} \in [0,127] \\
0 & \text{if } g_{ij} \in [128,255]
\end{cases}
\]  

(1)

If \( g_{ij} \in [0,127] \), it means the point in the \( i \)-th row and \( j \)-th of the image is white, then the corresponding element in the 0-1 matrix \( g_{1ij} = 1 \). If \( g_{ij} \in [128,255] \), it means the point in the \( i \)-th row and \( j \)-th of the image is black, then the corresponding element in the 0-1 matrix \( g_{1ij} = 0 \).

When given an image, this paper gets the outline and then turns it into a discrete one by proposing an operator template. After that, we could easily acquire the initial location of the drones on the sky. In Fig.1, there are \( 215^2 \) pixel points in images. If we take each pixel point as a 0-1 variable, there are too many variables in the optimal model which will bring a big trouble in solving process. To get the initial number of drones of an image in a light show and it is made up of pixels, this paper puts the image in a two-dimension coordinate and then can describe this image with a function \( f(i,j) \), where \( f \) is the gray value while \( (i,j) \) shows the coordinate of a pixel in the image. By referring to some literature [4-6] about image processing, this paper defined an operator as follow to discretize the image:

\[
f(i,j) = \begin{cases} 
0, & (i,j) = \left( \frac{n-1}{2}, \frac{n-1}{2} \right) \\
1, & \text{otherwise}
\end{cases}
\]  

(2)

where \( n \) is the size of the operator, and need to find a center section in this operator, so \( n \) is odd; \( f(i,j) \) is the gray value of the pixel of coordinate \( (i,j) \); Gray value equals 1 represents white, while 0 represents black.

To discretize the image with the operator, this paper uses the following algorithm and addresses the process in three steps:

Step1: using the operator to traverse the image from left to right first and from top to bottom then.

Step2: If the gray value in the center section not equals one, making it zero and then the \( n \times (n-1) \) neighborhoods one.

Step3: After traversing the whole image, we can get the discrete outline of the image and then the number of drones can be initially statistics.

From the above steps, we can see that the closest distance between two drones is greater than or equal to \((n-1)/2\), so the distance between two adjacent drones must exist in the same short outline of the image rather than different outlines. With the number and initial location of drones of an image, in order to determine the size of the show, we set the distance between drones and enhance that the closest distance between two drones are greater than the safe distance 1.5m. It means to zoom in the
whole image with a proportion of 1.5. Therefore, the length $L$ and width $W$ of the image can be easily calculated.

**Flight Strategy Based on the 0-1 Programming**

As the requirement of each image is determined, this paper establishes a 0-1 integer programming to figure out where the drone used in the former image go in the adjacent image. At first, we need to calculate the distance matrix $D = (d_{ij})_{n \times n}$, which include the distances of the points in the former image and the points of the adjacent image. $d_{ij}$ is the distance of position $i$ (in the former image) and position $j$ (in the adjacent image); $n$ is the number of the drones we need to arrange.

Next, set a 0-1 integer matrix to record the position change of each drone $Z = (z_{ij})_{n \times n}$. And for the value of the matrix, we define it as follows:

$$
z_{ij} = \begin{cases} 
0, & \text{if the drone of position } i \text{ do not move to position } j \\
1, & \text{if the drone of position } i \text{ need to move to position } j
\end{cases} \tag{3}
$$

What we need to do last is the set the target and the limit of the 0-1 programming:

$$
\min f = \max \{d_{ij}z_{ij}\}
$$

$$
\sum_{i=1}^{n} z_{ij} = 1, \text{ for } j = 1, 2, \ldots, n \tag{4}
$$

$$
\sum_{j=1}^{n} z_{ij} = 1, \text{ for } i = 1, 2, \ldots, n
$$

$$
z_{ij} \in \{0, 1\}
$$

These two constraint condition demand that the drone involved must go to one position of the latter image from the position of the former image.

In this way, we can figure out the best arrangement to take the least time for drones’ position changing during the light show.

In above model, we have determined each drone’s position in the displays. Apparently, the best path for the drone from point A to point B is going along the straight line which contacts the two points. If we arrange the path of each drone like this, of course, there may be the possibility of causing the collision. Apparently, it’s impossible for two drones to come into collision in the real 3D space if they do not meet each other during the position changing in the plane [7-9]. As a result, we just need to judge that whether the drones meet others or not in the plane. In this way, we can effectively simply the problem.

This paper developed a way figure out the drones which have the possibility of causing the collision. Then, we will show our method by taking two drones for example.

We set the first drone fly from the position $a$ to the position $a'$ while setting another one fly from the position $b$ to the position $b'$ in the plane. Combining the safety consideration (1m), we draw two segments on the two sides of the segment $aa'$ as $A_1A_1'$ and $A_2A_2'$. The area between the two segments is the ‘dangerous area’. Do the same operation with another drone, and we can get the segment $B_1B_1'$ and the segment $B_2B_2'$.

After all the preparation work has been done, we give the judgement as follows:

If there is no intersection point between the segment $A_1A_1'$ and the segment $B_1B_1'$, while there is no intersection point between the segment $A_2A_2'$ and the segment $B_2B_2'$, either. We can draw the conclusion that the two drones will not meet each other.

![Figure 3. The example of the remarks.](image-url)
If there is an intersection point \( P \) between the segment \( A_1A_1' \) and the segment \( B_1B_1' \) while there is no intersection point between the segment \( A_2A_2' \) and the segment \( B_2B_2' \).

As all the drones have been set to fly at the same speed, the length of the segment can reflect the time the drone used. On the other hand, we have assumed that the different drone set off at the same time. As a result, we can judge the collision in this way:

If \( |A_1P_1| = |B_1P_1| \), the two drones come into collision. If \( |A_1P_1| \neq |B_1P_1| \), the two drones is safe.

If there is an intersection point \( P_1 \) between the segment \( A_1A_1' \) and the segment \( B_1B_1' \) while there is also an intersection \( P_2 \) point between the segment \( A_2A_2' \) and the segment \( B_2B_2' \).

If \( (|A_1P_1| - |B_1P_1|)(|A_2P_2| - |B_2P_2|) > 0 \), the two drones is safe.

If \( (|A_1P_1| - |B_1P_1|)(|A_2P_2| - |B_2P_2|) \leq 0 \), the two drones come into collision.

Conclusion

In this paper, several models were proposed to provide a wonderful light show. Firstly, this paper defined an operator template to discretize the images and get the location of drones in each images. Then, this paper calculated the distance matrix between images and proposed the optimal flight path of the drones to minimize the maximum distance between the adjacent images. Finally, this paper discussed the safe flight path of drones to avoid collisions.

References


