UAV Collision Detection Algorithm based on Formula for Fast Calculation

Jung Kyu Park¹ and Jaeho Kim²*

¹Department of Computer Software Engineering, Changshin University, Korea
²Dept. of Aerospace and Software Engineering, Gyeongsang National University, Korea
*Corresponding author. E-mail: jaeho.kim@gnu.ac.kr

Article History: Received: 11 November 2020; Accepted: 27 December 2020; Published online: 5 April 2021

Abstract: Collision Avoidance System (CAS) is known as a pre-collision system or a forward collision warning system, and research has first begun as a vehicle safety system. In this paper, we propose an algorithm for collision detection of UAV. The proposed algorithm uses a mathematical method and detects the collision of the UAV by modeling it in a two-dimensional plane. Using the mathematical modeling method, it is possible to determine the collision location of the UAV in advance. Experiments were conducted to measure the performance and accuracy of the proposed algorithm. In the experiment, we proceeded assuming three environments and were able to detect an accurate collision when the UAV moved. By applying the algorithm proposed in the paper to CAS, many collision accidents can be prevented. The proposed algorithm detects collisions through mathematical calculations. In addition, the movement time of the UAV was modeled in a 2D environment to shorten the calculation time.

Keywords: Collision Detection; Intersection; Line equation; Independent variables; UAV

1. Introduction

A drone was used for the first time for military use, but recently the market is rapidly growing for agricultural, service, and leisure purposes. As the market has grown rapidly, many hobby and leisure drones that can be purchased at low cost have become popular. However, as many drones are used, many accidents are occurring. For example, drone battery explosions, fires, injuries caused by collisions with drones, and safety accidents such as falls and collisions due to deviation from the adjustment distance are rapidly increasing. For this reason, a Collision Avoidance System (CAS) is essential in drones or UAV (Unmanned Aerial Vehicle) [1-4].

CAS is known as a pre-collision system or a forward collision warning system, and research has first begun as a vehicle safety system [5-7]. CAS research has been conducted in various fields as well as in the automobile field, and research is rapidly progressing in the field of aircraft and UAV. Although there are various studies related to CAS, the basic algorithm of CAS is to create a safe travel path without causing a collision from a starting point to a target point. The CAS structure is composed of three elements: search, recognition, and avoidance. Among the elements, search and recognition are responsible for collision detection. And the evasion performs a collision avoidance mission. For UAVs to perform their CAS mission, it is necessary to identify and collaborate with the surrounding environment when first flying. At this time, the obstacle information of the environment can be collected in real time through the mounted sensor of the board [8-10].

Since the UAV moves in autonomous driving without a pilot, a powerful CAS must be used. UAVs are much cheaper to operate than human-controlled airplanes. It is also more suitable for performing dangerous or repetitive tasks that are difficult for humans to perform. Because of the above advantages, UAVs can be used for civilian tasks such as mapping and rescue, or for military missions such as patrol and search. To perform this specific task, multiple UAVs can be used at the same time, and each UAV must establish a mobile sensor network. In addition, UAVs must have communication, navigation, and collision avoidance functions [11-14].

In this paper, we propose an algorithm for collision detection of UAV. The proposed algorithm uses a mathematical method and detects the collision of the UAV by modeling it in a two-dimensional plane. Using the mathematical modeling method, it is possible to determine the collision location of the UAV in advance. An additional reason to use mathematical models is performance. When the UAV driving mode is changed from a simple navigation mode to a collision detection mode, real-time processing performance of collision detection is important. Using the proposed algorithm, it is possible to determine the time until the collision of the UAV, the travel distance, and the collision point.

2. Proposed Method

UAV collision detection algorithm proposed in this paper models UAVs moving in three dimensions in two dimensions and expresses them with equations. Some assumptions are made for fast computation of the algorithm. Because it aims to detect collisions between UAVs, the UAV is moving and the initial and final positions are not considered. Figure 1 shows the situation in which collision occurs in two dimensions.
2.1 Proposed Method

In this section, we describe the equations that form the basis of the algorithms we want to present in our study. This formula is used to detect collisions and calculate collision points.

![Figure 1. Collision point of moving 2UAVs](image)

2.2 Collision Point between UAVs

As shown in Figure 2, when the UAV is moving in a straight line, the equation of the straight line can be obtained using Equation (3).

\[ P_A = m_A \cdot (a_2 - a_1) + a_1 \]
\[ P_B = m_B \cdot (b_2 - b_1) + b_1 \]

\[ m_A \] and \[ m_B \] of equations (4), (5) mean the scale of point \( P_A \) and \( P_B \).
To detect collisions between UAVs, the coordinate values of the two UAVs must be continuously monitored. That is, when two UAVs have the same coordinates \((P_A = P_B)\), a collision occurs. Assuming that the coordinates of the two UAVs are the same, we can organize the expressions by \(P_A.x = P_B.x\) and \(P_A.y = P_B.y\). Summarizing the previous equation, the equation is (6).

\[
\begin{align*}
m_A &= (a_2.x - b_1.x)b_2.y - b_1.y + b_1.y \\
&= (a_2.y - a_1.y)(b_2.x - b_1.x) - (a_2.x - a_1.x)(b_2.y - b_1.y)
\end{align*}
\]

Using the two equations in Equation 6, \(m_A\) and \(m_B\) can be obtained. In Equation 7, \(m_A\) and \(m_B\) are summarized.

\[
\begin{align*}
m_A &= (a_2.x - a_1.x)(b_1.y - a_1.y) - (a_2.y - a_1.y)(b_1.x - a_1.x) \\
&= (a_2.y - a_1.y)(b_2.x - b_1.x) - (a_2.x - a_1.x)(b_2.y - b_1.y)
\end{align*}
\]

In order to determine whether two UAVs collide, we need to make sure that both UAVs converge to one point. That is, it is possible to determine whether collision occurs by evaluating whether, \(m_A\) and \(m_B\) are values between 0 and 1. That is, when the two conditions in Equation (8) are satisfied, the collision point occurs in the UAV moving line.

\[
\begin{align*}
0 < m_A &< 1 \\
0 < m_B &< 1
\end{align*}
\]

Although the predicted point of collision can be predicted by Equation (8), additional considerations must be taken to actually detect the collision. An additional consideration is the travel time of the UAV. The time at the current location of the two UAVs is compared, and if the difference is smaller than the threshold, collision avoidance should be performed. It is assumed that the current time of each UAV is \(t_A\) and \(t_B\). When the difference between the two UAVs' current time is less than the threshold value, a collision occurs between the UAVs. At this time, the time difference must be less than the threshold value \(t_H\) and greater than 0. In this case, the threshold value should consider the operating environment of the UAV and the hardware characteristics of the UAV.

2.3 Collision Detection

Algorithm 1 describes a collision detection technique based on the section 2.2. This algorithm is performed when the UAV is on the move. Periodically, \(m_A\) and \(m_B\) values are obtained, and when these values are between 0 and 1, \(t_A\) and \(t_B\) values are additionally obtained. When the difference between the values of \(t_A\) and \(t_B\) is less than \(t_H\), a collision avoidance operation is performed in anticipation of a collision.

Algorithm 1: Collision Detection

```
1  let, \(t_H\) is a threshold of time
2  while moving UAVs do
3    calculate the \(m_A\) and \(m_B\);
4    if ( \(0 < m_A < 1\) && \(0 < m_B < 1\) ) then
5      get current time of two UAVs, \(t_A\) and \(t_B\);
6      if ( \(0 \leq |t_A - t_B| < t_H\) ) then
7        perform the collision avoidance;
8    end
9  end
10 end
```

3. Results and Discussion

In order to verify the proposed algorithm, a simulation experiment was conducted. The experiment was performed assuming that the UAV was moving in a 2D environment. The experiment was divided into three categories. 1) When there is no collision point, 2) When an intersection occurs and a collision is expected. 3) When an intersection occurs but there is no collision. As such, experiments were performed based on three scenarios, and the detailed experiment environment is shown in Table 1.

<table>
<thead>
<tr>
<th>UAV 1</th>
<th>Category 1</th>
<th>Category 2</th>
<th>Category 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start position</td>
<td>(-10, 20)</td>
<td>(-10, 0)</td>
<td>(-10, 0)</td>
</tr>
<tr>
<td>Destination position</td>
<td>(10, 20)</td>
<td>(10, 0)</td>
<td>(10, 0)</td>
</tr>
</tbody>
</table>

Table 1: Data set for experiment
Based on Table 1, experiments were performed by category, and as a result, Figures 3, 4, and 5 were obtained. Based on Table 1, experiments were performed by category, and as a result, Figures 3, 4, and 5 were obtained. In Figure 3, the two UAVs move horizontally at the same speed, so there are no intersections and collisions. That is, when $m_A$ and $m_B$ are calculated by Equation 7, the denominator becomes 0, so $m_A$ and $m_B$ cannot be obtained. This means that when two UAVs move, they do not intersect at the time $t_1$ and $t_2$.

Figure 4 shows the results of Category 1. Like Category 1, two UAVs are moving at the same speed. At this time, if the values of $m_A$ and $m_B$ are obtained using Equation (7), each is 0.5, which is greater than 0 and less than 1. It can be seen that a collision occurs at time $t_1$ at (0,0). Figure 5 shows the results of Category 3. The two UAVs are moving at different speeds. At this time, if the values of $m_A$ and $m_B$ are obtained using Equation 7, each is 0.5, which is the same as the result in Figure 3. Calculating the intersection point of the UAV is (0,0), but no collision occurs. That is, because two UAVs move at different speeds, each UAV is sufficiently far away at the time $t_1$.

<table>
<thead>
<tr>
<th>UAV 2</th>
<th>Speed (m/s)</th>
<th>(40km/h)</th>
<th>(40km/h)</th>
<th>(20km/h)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Start position</td>
<td>(-10, 10)</td>
<td>(-10, -10)</td>
<td>(-10, -10)</td>
</tr>
<tr>
<td></td>
<td>Destination position</td>
<td>(10, 10)</td>
<td>(10, 10)</td>
<td>(10, 10)</td>
</tr>
<tr>
<td></td>
<td>Speed (m/s)</td>
<td>40km/h</td>
<td>40km/h</td>
<td>40km/h</td>
</tr>
</tbody>
</table>
4. **Conclusions**
In this paper, we propose an algorithm to detect collisions that may occur when UAVs move. The proposed algorithm detects collisions through mathematical calculations. In addition, the movement time of the UAV was modeled in a 2D environment to shorten the calculation time. To use the collision detection algorithm, two values require the current location and time of the UAV. In order to evaluate the proposed algorithm, a simulation experiment was conducted, and as a result, it was possible to accurately determine whether or not a collision occurred. Future studies will further refine the UAV crash situation and add more variables for calculation.

5. **Acknowledgements**
This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2018R1C1B5046282).

6. **References**