UKRAINIAN CATHOLIC UNIVERSITY

BACHELOR THESIS

Multi-camera visual collision avoidance for micro aerial vehicles

Author:
Mykola MORHUNENKO

Supervisor:
Ing. Matouš VRBA

A thesis submitted in fulfillment of the requirements for the degree of Bachelor of Science

in the

Faculty of Applied Sciences
Department of Computer Sciences

Lviv 2022

Declaration of Authorship

I, Mykola MORHUNENKO, declare that this thesis titled, "Multi-camera visual collision avoidance for micro aerial vehicles" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

"Science, my lad, is made up of mistakes, but they are mistakes which it is useful to make, because they lead little by little to the truth."

Jules Verne

UKRAINIAN CATHOLIC UNIVERSITY

Faculty of Applied Sciences
Department of Computer Sciences

Bachelor of Science

Multi-camera visual collision avoidance for micro aerial vehicles

by Mykola MORHUNENKO

Abstract

We live in a twenty first century - time of extremely fast developing of all electronic devices. Each month we see some brake-through in such directions as microchips modeling, flying vehicles developing, quantum computing and space exploration, bioengineering and medicine.

in the thesis I would like to focus on micro aerial vehicles as one of the most perspective development directions and interesting personally for me. During my internship in the MRS Group I was working a lot with computer vision, and I see it as quite promising direction.

Specifically, visual collision avoidance is one of the relevant topics that is being actively researched nowadays. Drones with this feature becomes more preferable - they are safer, can last longer and easier to control. Visual collision avoidance is much less expensive than Lidars.

:TODO

Acknowledgements

I would like to thank my supervisor Ing. Matouš Vrba and all CTU MRS Group for the opportunity to have an internship there for last one and a half year, all of them was kind, they helped me a lot with advice's and sharing their experience, CTU FEE for interesting and useful for this thesis courses. I want to admire that nothing my study will not take place without my small family - Ukrainian Catholic University, especially Applied Science Faculty, all teachers and deanery for saving all students from all the worst sides of Ukrainian education system and providing only the best quality education without corruption and plagiarism.

Especially I want to thank all defenders of my Motherland, The Armed Forces of Ukraine, who protect the whole Europe during the russo-Ukraian war at the cost of their own lives to make it possible for all of us to live in peace and for me to write this thesis.

Finally, I would like to thank my family who supported me during whole my life, my mother Svitlana and my father Roman.

Contents

List of Figures

List of Tables

List of Abbreviations

UAV	**U**nmanned **A**erial **V**ehicle
MAV	**M**icro Unmanned **A**erial **V**ehicle
ROS	**R**obotic **O**perating **S**ystem
DoF	**D**egree **o**f **F**reedom

List of Symbols

f	focal length	px

Chapter 1

Introduction

Micro unmanned aerial vehicles (MAVs) recently saw a rise in usage across various fields. Drones are already wide used in cinematography[1] and advertising[2], In Ukraine they are very helpful in farming (to apply pesticides to fields)[3]. City emergency departments use UAVs - firefighters can use them to see and evaluate the situation from the sky, localise the source of fire and deal with that[4], sometimes it even can have some fire-extinguishing capsules as projectiles[5]. They are also quite popular in military industry.

The inspiration for this project was taken from DJI obstacle avoidance technology introduced with the release of the DJI Mavic 3 drone[6] on fifth November 2021. Despite the fact that the idea is old, neither DJI nor MRS nor other research groups have a well-developed visual obstacle avoidance system, the best for now can be Skydio obstacle avoidance system , so this direction is very perspective for researchers. Many drones available for sale are costly, and even a well-trained pilot is afraid of crashing. At the same time, autonomous drones are more predictable than a human pilot, behave acording to algorithms and can react much faster, but only if they have a well-designed system running on board, so obstacle avoidance for autonomous MAVs will be both more challenging and more critical in future trends.

1.1 Problem definition

While obstacle avoidance considers static objects, collision avoidance is related to averting crash with moving objects like other MAVs, cars or people. It is a complicated task but more relevant to multi-robot systems, because during interactions between robots they should not brake each other.

The goal of this thesis is to implement an obstacle avoidance system, and expand it to collision avoidance system for autonomous MAVs driven by a Robotic operating system (ROS)[7] using the MRS UAV system [1] [8].

The problem solution can be devided into several steps: firstly it is necessary to model such device, assemble and calibrate it, then find a pointcloud using a structure from motion algorithm for each camera in a system and find moving objects using the fact of overlaping zones for each camera pair. Then use some algorithm

[1] Coptrz, "How drones are used in big-budget films
[2] Bangkokpost, "The future of advertising could be drones"
[3] DroneUA
[4] Fire Fighting Drones
[5] Autonomous Firefighting Inside Buildings by an Unmanned Aerial Vehicle
[6] DJI Mavic 3
[6] Skydio autonomy
[7] ROS home page
[8] MRS UAV system

for path planing to correct and update the previous path. As for now, the most complecated task is to find an obsticle using a visual method, so this thesis focuses on this particular part of a problem.

1.2 Related Works

There are several obstacle avoidance sensors used by various MAVs: stereo vision [2], depth cameras (as Intel RealSense), monocular vision [3], lidar (2d or 3d) [4], sonar (ultrasonic), time of flight sensors, also combinations of them can be used. In [5] the sensor fusion of ultrasonic and infrared sensors is presented.

Each of them has its pros and cons. 3d lidars are extremely expensive but the most efficient for today; 2d lidars are used for small ground vehicles, but not suitable for most tasks for MAVs (because a car can be modelled as a 2 DoF system, while MAV always has 6 DoF), depth cameras are relatively expensive too, ultrasonic and infrared sensors both have distance limits and other minor issues. Overall, stereo vision is the most promising approach for the nearest future.

Most articles uses stereo pair of two parallel cameras looking in the same direction (classical stereo pair) [6, 7, 8] or deep learning approaches [9, 10, 11, 12]. Realtime multi-camera feedback control system is introduced in [13], but this solution does not imply that drone can fly in any direction, only forward moving counted, still this work is incredibly inspiring.

Chapter 2

Basic concepts

2.1 Homogenous coordinate system

In a projective geometry, homogenous coordinate system is used in the same way as Cartesian coordinates are used in Euclidian geometry. To transform a point $x = (u, v)$ from cartesian coordinates to homogenous, simply add the third coordinate 1: $x = (u, v, 1)$. Homogenous coordinates are used to simplify the 2D transformation operations:

Scale

$$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \end{bmatrix}$$

Rotation

$$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} cos(\theta) & -sin(\theta) \\ sin(\theta) & cos(\theta) \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \end{bmatrix}$$

Translation

$$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

To aply any transformation we need to make a sequence of matrix multiplications, but this is not the case with translation - we need an addition operation for that. Here is how all this operations looks like in a projective geometry:

Scale

$$\begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \\ 1 \end{bmatrix}$$

Rotation

$$\begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} = \begin{bmatrix} cos(\theta) & -sin(\theta) & 0 \\ sin(\theta) & cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \\ 1 \end{bmatrix}$$

Translation

$$\begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \\ 1 \end{bmatrix}$$

So in homogenous coordinate system all 2d tansformations can be combined and expressed as matrix multiplications.

FIGURE 2.1: Parallel lines intersection

2.1.1 Infinity

In Euclidian geometry parallel lines are the lines that have no intersection point. In projective geometry, parallel lines are intersecting in a point x at infinity. How is it possible? In the image Figure 2.1 let $l_r = Pr_1 \cap Pr_2$ and $l_l = Pl_1 \cap Pl_2$. In 3D world we know that lines $l_l \parallel l_r$, but after projection on the image plane Π, we can calculate - in this case even see - the intersecting point P_∞.

Both line and point in a homogenous coordinates can be expressed as vector of three numbers, but if in case of a point homogenous $x = (x, y, z)^T$ will be $m = (\frac{x}{z}, \frac{y}{z})^T$, line $l = (a, b, c)^T$, where a, b, c are parameters of a line equation $ax + by + c = 0$. In general case point at infinity is called an *ideal point* and it is not seen on the image, it's coordinates can be expressed as $x_\infty = (u, v, 0)^T, \{u, v\} \neq 0$. Same about line - such line is called *ideal line* and it's coordinates are $l = (0, 0, c), c \neq 0$. In algebraic representation, both ideal point and line are lying at the plane Π_0 (Figure 2.2, green lines).

2.2 Pinhole camera model

Pinhole camera - or a canonical perspective camera model - is a model of a simple camera without optics. The very first example is a camera obscura - a dark room

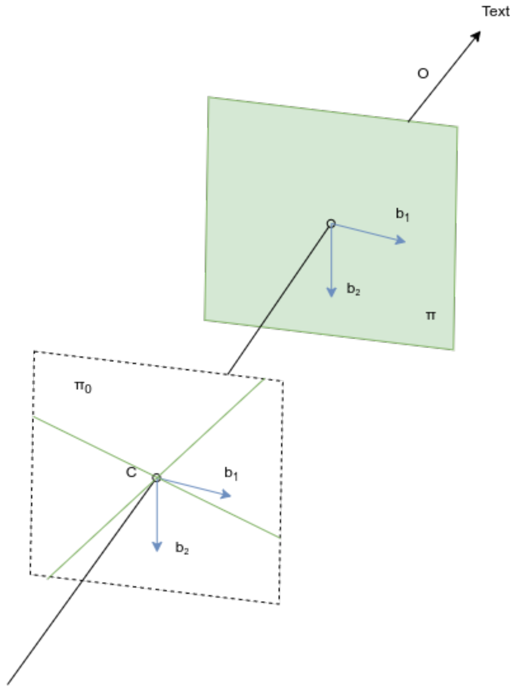

FIGURE 2.2: Scheme of homogenous coordinates

with a small hole, through which the image from outside is projected on the oposite wall. This model can be used to express camera geometry with field of view angles less than 180°.

2.2.1 Camera coordinate system

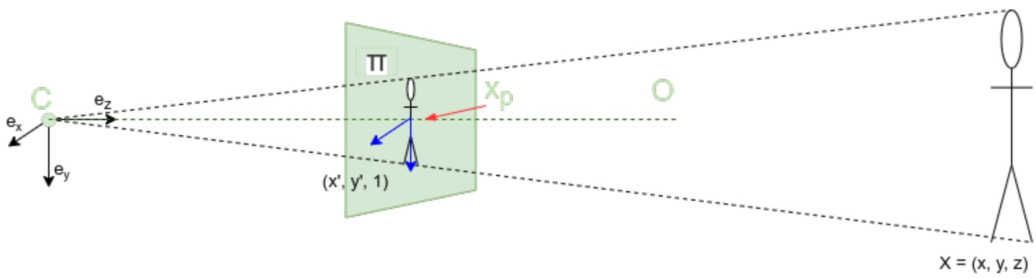

FIGURE 2.3: The pinhole camera model working scheme

In physical implementation of Obscure camera the projective plane is on the oposite side from the Projection center (or Camera center C in pinhole camera model), the image is reversed and mirrored, but in most computer vision literature authors assume that it is on the same side as object (see Figure 2.3). In Figure 2.3 we are looking through a camera with camera center C in a coordinate system with origin at C and basis vectors (e_x, e_y, e_z) on a human. Each point $X = (x, y, z)^T$ in a world coordinate system has it's projection $x_p = (x', y', 1)^T$ on a plane Π which is located on a distance 1 from a camera center (Figure 2.4). Optical axis O is a ray perpendicular to plane Π, and on the image the point $O \cap \Pi = x_p$ is a center of the image, see Figure 2.5).

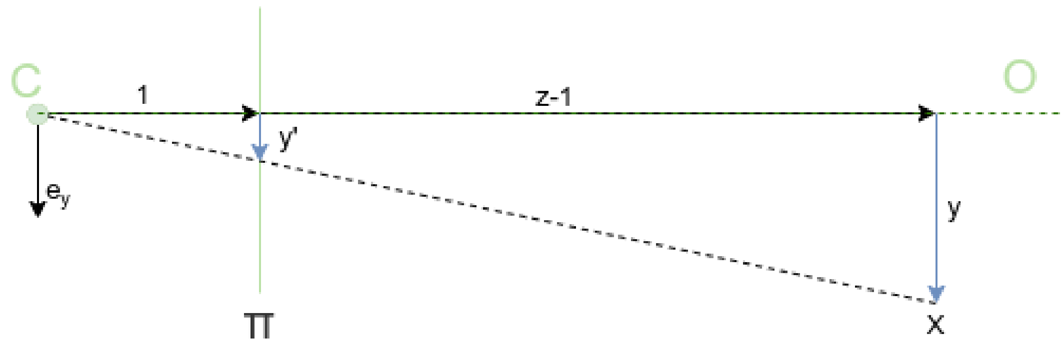

FIGURE 2.4: The pinhole camera model, y-z plane

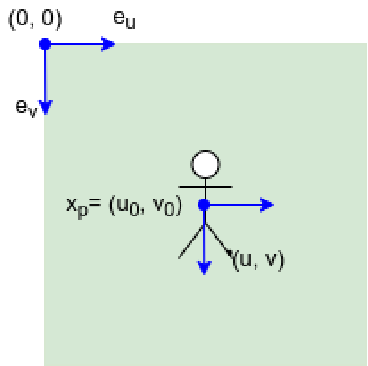

FIGURE 2.5: The pinhole camera model, x-y plane

2.2.2 Camera calibration matrix

Camera calibration matrix - a matrix that includes camera *intrinsic* parameters - pixel size (e_u and e_v) and pixel skew angle (θ), as on Figure 2.6, pixel aspect ratio **a** and principle point coordinates $x_p = (u_0, v_0)$.

$$K = \begin{bmatrix} af & -afcot(\theta) & u_0 \\ 0 & f/sin(\theta) & v_0 \\ 0 & 0 & 1 \end{bmatrix} \qquad units : [f] = px, [u_0] = px, [v_0] = px, [a] = 1$$

Where f is a focal length used to convert world length ratios to pixels.

In a modern world, every digital camera has a calibration matrix with a square pixel, so in most cases camera matrix looks like:

$$K = \begin{bmatrix} f & 0 & u_0 \\ 0 & f & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

2.2.3 Projection matrix

To translate a point from a world coordinate frame to image coordinate frame, Image projection matrix P is used. The canonical projection matrix P_0 assumes that the camera is in the world coordinate center and that the calibration matrix $K = \mathbf{I}$

$$P_0 = \begin{bmatrix} \mathbf{I} & | & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

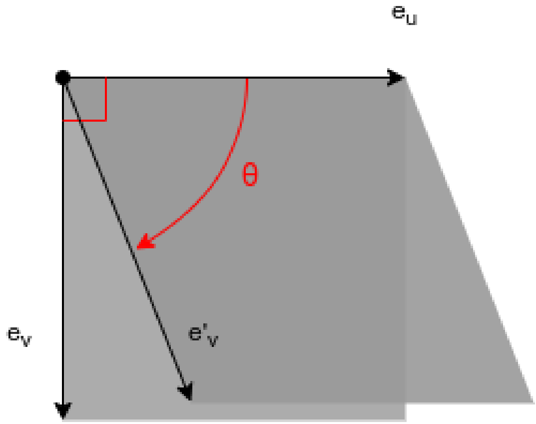

FIGURE 2.6: Scheme of pixel, changing the image (inner) reference frame

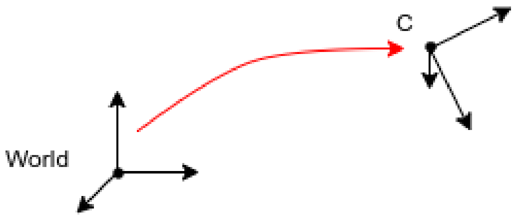

FIGURE 2.7: Changing the world (outer) reference frame

But this case is degenerate. As far as each camera is different, canonical projection matrix is never used, instead image projection matrix P is used, with applied calibration matrix K to transform canonical P_0 to perspective P:

$$P = \mathbf{K}\begin{bmatrix}\mathbf{I} & | & 0\end{bmatrix} = \begin{bmatrix} f & 0 & u_0 & 0 \\ 0 & f & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

But not always the world coordinate center is located at point C Figure 2.7. Usually it is rotated using a rotation matrix R and translated on vector t where R is a $3x3$ matrix with $det(R) = 1$ and $R^{-1} = R^T$. So in general case:

$$P = K\begin{bmatrix}\mathbf{R} & | & \vec{\mathbf{t}}\end{bmatrix} = K\begin{bmatrix}\mathbf{R} & | & -\mathbf{RC}\end{bmatrix}$$

where C is quite often used as a camera position in a world reference frame. So matrix P has 6 intrinsic parameters: 3 Euler angles and 3 translation components.

2.2.4 Projection equation

Image point $m = (u, v)^T$ can be obtained from a point X using P

$$\lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = P \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

$$\lambda \begin{bmatrix} \vec{m} \\ 1 \end{bmatrix} = P \begin{bmatrix} \vec{X} \\ 1 \end{bmatrix}$$

Where $\lambda \neq 0$

2.3 Epipolar geometry

2.3.1 Skew-symmetric 3x3 matrix

From [14], p.581.

Skew-symmetric or antisymetric matrix is such matrix $[b]_\times$ that $[b]_\times^T = -[b]_\times$. For vector $b = (b_1, b_2, b_3)^T$:

$$[b]_\times = \begin{bmatrix} 0 & -b_3 & b_2 \\ b_3 & 0 & b_1 \\ -b_2 & b_1 & 0 \end{bmatrix}$$

This matrix has some important properties, but the most important in this thesis - it generalizes a cross product as a matrix multiplication

$$a \times b = [a]_\times b$$

2.3.2 Epipolar geometry

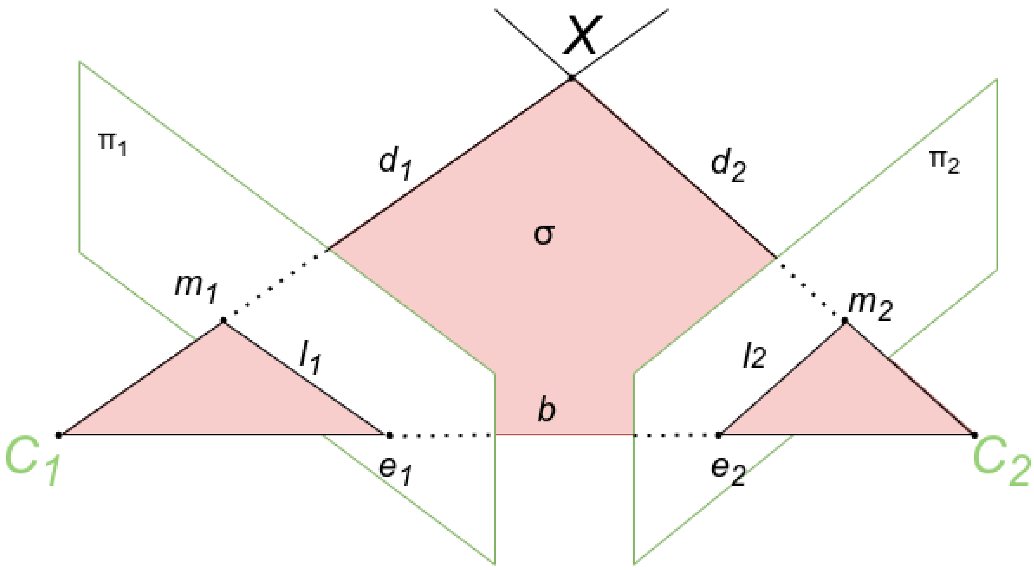

FIGURE 2.8: Epipolar geometry scheme

Figure 2.8 shows a scheme of two cameras with different camera centers C_1 and C_2 connected with a base line $b = C_2 - C_1$. Both of them are seen some 3d point X. This point projections are m_1 and m_2 respectivly. Points C_1, C_2 and X form an *epipolar plane* σ. $\sigma \cap \pi_1 = l_1$ and $\sigma \cap \pi_2 = l_2$ are images of *epipolar lines*. Epipolar line l_1 passes through *epipole* e_1, where $\lambda[e_1|1]^T = P_2C_1$, respectivly $\lambda[e_2|1]^T = P_1C_2$.

2.3.3 Epipolar constraint

Having a set of two cameras the relationship between them and constraints on them can be expressed by two new matrices: *Essential matrix* $E \in \mathbb{R}^{3x3}, rank(E) = 2$

$$E = R_2[C_2 - C_1]_\times R_1^T = [-t_{21}]_\times R_{21}$$

and *Fundamental matrix* $F \in \mathbb{R}^{3x3}, rank(F) = 2$

$$F = K_2^{-T} R_2 [C_2 - C_1]_\times R_1^T K_1^{-1} = K_2^{-T}[-t_{21}]_\times R_{21} K_1^{-1} = K_2^{-T} E K_1^{-1}$$

where $R_{21} = R_2 R_1^T$ is a relative camera rotation and $t_{21} = -R_2 b = t_2 - R_{21} t_1$ is a relative camera translation. The translation t_{21} is lost since E is homogenous. Epipolar constraint looks like

$$\begin{bmatrix} m_2 & | & 1 \end{bmatrix} F \begin{bmatrix} m_1 \\ 1 \end{bmatrix} = 0$$

F maps points to lines, such as

$$\lambda l_1 = F^T \begin{bmatrix} m_2 \\ 1 \end{bmatrix}$$

$$\lambda l_2 = F \begin{bmatrix} m_1 \\ 1 \end{bmatrix}$$

Also some other properties of F matrix:

$$F \begin{bmatrix} e_1 \\ 1 \end{bmatrix} = F^T \begin{bmatrix} e_2 \\ 1 \end{bmatrix} = 0$$

Chapter 3

Calibration

3.1 Camera calibration

Camera calibration - computing the camera intrinsic matrix.

3.2 Stereopair calibration

Chapter 4

Conclusion

Bibliography

[1] T. Baca, M. Petrlik, M. Vrba, V. Spurny, R. Penicka, D. Hert, and M. Saska, "The MRS UAV system: Pushing the frontiers of reproducible research, real-world deployment, and education with autonomous unmanned aerial vehicles," *Journal of Intelligent & Robotic Systems*, vol. 102, Apr. 2021.

[2] B. Ruf, S. Monka, M. Kollmann, and M. Grinberg, "Real-time on-board obstacle avoidance for uavs based on embedded stereo vision," 2018.

[3] L. Mejias, S. McNamara, J. Lai, and J. Ford, "Vision-based detection and tracking of aerial targets for UAV collision avoidance," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, Oct. 2010.

[4] S. Ramasamy, R. Sabatini, A. Gardi, and J. Liu, "LIDAR obstacle warning and avoidance system for unmanned aerial vehicle sense-and-avoid," *Aerospace Science and Technology*, vol. 55, pp. 344–358, Aug. 2016.

[5] R. Rambabu, M. R. Bahiki, and S. Azrad, "MULTI-SENSOR FUSION BASED UAV COLLISION AVOIDANCE SYSTEM," *Jurnal Teknologi*, vol. 76, Sept. 2015.

[6] Y. Yu, W. Tingting, C. Long, and Z. Weiwei, "Stereo vision based obstacle avoidance strategy for quadcopter UAV," in *2018 Chinese Control And Decision Conference (CCDC)*, IEEE, June 2018.

[7] H.-Y. Lin and X.-Z. Peng, "Autonomous quadrotor navigation with vision based obstacle avoidance and path planning," *IEEE Access*, vol. 9, pp. 102450–102459, 2021.

[8] Y. Xiao, X. Lei, and S. Liao, "Research on UAV multi-obstacle detection algorithm based on stereo vision," in *2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, IEEE, Mar. 2019.

[9] S. Back, G. Cho, J. Oh, X.-T. Tran, and H. Oh, "Autonomous UAV trail navigation with obstacle avoidance using deep neural networks," *Journal of Intelligent & Robotic Systems*, vol. 100, pp. 1195–1211, Sept. 2020.

[10] P. Fraga-Lamas, L. Ramos, V. Mondéjar-Guerra, and T. M. Fernández-Caramés, "A review on IoT deep learning UAV systems for autonomous obstacle detection and collision avoidance," *Remote Sensing*, vol. 11, p. 2144, Sept. 2019.

[11] B. Park and H. Oh, "Vision-based obstacle avoidance for UAVs via imitation learning with sequential neural networks," *International Journal of Aeronautical and Space Sciences*, vol. 21, pp. 768–779, Feb. 2020.

[12] J. Roghair, A. Niaraki, K. Ko, and A. Jannesari, "A vision based deep reinforcement learning algorithm for UAV obstacle avoidance," in *Lecture Notes in Networks and Systems*, pp. 115–128, Springer International Publishing, Aug. 2021.

[13] D. He, , H.-M. Chuang, J. Chen, J. Li, and A. Namiki, “Real-time visual feedback control of multi-camera UAV,” *Journal of Robotics and Mechatronics*, vol. 33, pp. 263–273, Apr. 2021.

[14] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2 ed., 2004.