iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** V **Month of publication:** May 2022

DOI: https://doi.org/10.22214/ijraset.2022.43588

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue V May 2022- Available at www.ijraset.com

Monocular Visual SLAM with 256-Bit Encryption on UAV

Prof. Ayushi Jaiswal[1], Sahil Nagmote[2], Sumit Khobragade[3], Vaibhav Rajsewati[4]
[1, 2, 3, 4]*Electronics and Telecommunications Engineering Department, S.B Jain Institute of Technology, Management and Research*

Abstract: ***Monocular Visual Slam with 256-Bit Encryption on UAV is based on the concept of Fault Tolerance Control System which is very crucial for the safety of critical systems such as quadrotors. Although, this method relies on external sensors, such as GPS or motion capture systems for state estimation. In this project, we propose the algorithm that combines fault tolerance control and onboard vision-based state estimation to achieve position control of a quadrotor subjected to complete failure of one rotor. It will use the video feed as input to create 2D pose estimation graph which will help it to pinpoint its location. In order to perform the VIO i.e., Visual Inertial Odometry, we need to combine the IMU sensor reading with the data coming from the monocular camera. To use VIO, PX4 Enabled flight controller needs to communicate with the companion computer via UART pin using MAVROS. Due to this, it will be an effective device in GPS denied environment. Its design will also allow the system to follow its set trajectory even after the rotor failure. And at last, the digital data transmission will take place in an encrypted format with AES-256 encryption to back up the data securely. We believe that our approach will render autonomous quadrotors safer in both GPS denied or degraded environment.***
Keywords: AES-256, Encryption, Visual SLAM, Visual Inertial Odometry, IMU

I. INTRODUCTION

In this fast-changing tech world, the need of using UAVs is increasing which led to questioning the reliability of sensors used in UAVs. one of the most important sensors that help UAVs to function autonomously is GPS & compass module.

As we talk about the reliabilities of UAVs, we must prevent this expensive piece of hardware from crashing. This is where the concept of a fault tolerance control system comes in which if one rotor of a quadcopter fails or malfunctions then the other three-rotor should be able to maintain the position of the drone without losing its trajectory. How the fault tolerance algorithm works are we need to prioritize the control axis of the drone i.e., throttle, pitch, yaw, and roll. The horizontal channel which consists of pitch and roll is given the highest priority, the thrust channel which is throttle control is after and then the least priority is given to the yaw channel. Generally, fault tolerance can be done by rotor redundancy, or having a GPS can do the job.

But it becomes difficult to do fault tolerance in GPS denied or GPS degraded environments. That is why it is essential that we find a solution which is a VISUAL SLAM using Mono RGB-D and a stereo camera.

All the digital data transmission taking place shall pass through an encapsulating code written in Crypto. The video signals that the UAV will transmit from the monocular camera will be unencrypted and thus will be vulnerable to the subject of getting remote access from a different device. To prevent that, we will add some alphanumeric value, which is also known as salt, which will give us an encrypted signal value. This can only be read by the destination computer which will have the key to decrypt the signal to view it. The encryption will be of the AES 256-Bit encryption Standard which is considered to be the strongest encryption.

II. ORB-SLAM2

Concept of SLAM can be really essential when it comes to the precision mapping and ability to work without GPS. Generally, SLAM uses laser-based sensors like Lidar or acoustic based sensor, since lidars can be bit heavy and won't be efficient aerodynamically we could use camera on UAV. This can provide two purposes, first is the visual sensor and second could be surveillance or just a cockpit view from the drone.

Here we have used a 450mm base drone with raspberry pi as companion pc mounted on drone with a generic Logitech webcam connected directly to the raspberry pi. In order to use visual SLAM more precisely with a quadcopter we could use visual inertial odometry (VIO). The heart of this whole visual slam is ORB-SLAM2 (an open-source slam system). ORB-SLAM2 consists of three important parallel threads: tracking, local mapping and loop closing, a fourth thread can be created to perform full bundle adjustment after loop closure.

©IJRASET: All Rights are Reserved

ORB-SLAM2 performs motion bundle matching to minimize errors in placing each feature in the correct position. This is also known as minimizing reprojection errors. After this local mapping is done and these maps are optimized using ICP (iterative closest point), this gives the most probable pose of the camera in 3D space. Then use the pose graph optimization to correct the accumulated deviations and close the loop.

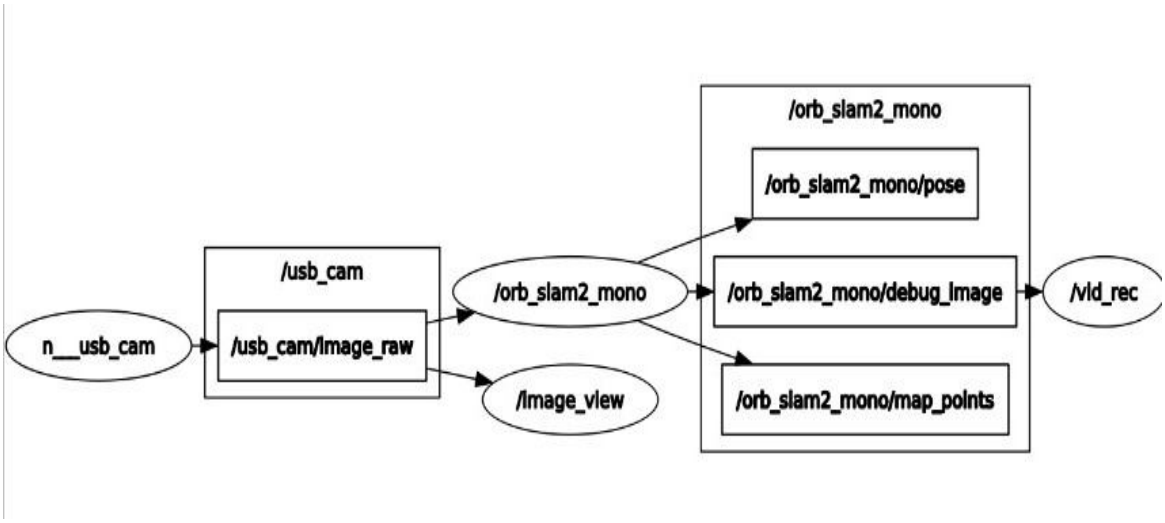


Fig. 1 – Block Giagram of ORB-SLAM2

DBoW2 is an open-source C++ library for indexing and converting images into bag-of-word representation. This DBoW2 is used for relocalization, to prevent failure of tracking. The system maintains a graph consisting of a keyframe as a node and edge between keyframe exists if they share at least 15 common map points. This map connects any two keyframes, respecting the commonalities and the minimum spanning tree that connects all the keyframes.

A. Tracking in Visual SLAM

ORB features are often used in computer vision library like OpenCV to compare two frames using similar and identical features from the two frames. These ORB features are used to track mapping and location detection tasks on this system. The ORB provides excellent camera gain and exposure, as well as excellent immutability to changing lighting. Even though ORB features detect a lot of features in a particular frame but not all the features contribute to accurate estimation of the camera poses and the map.

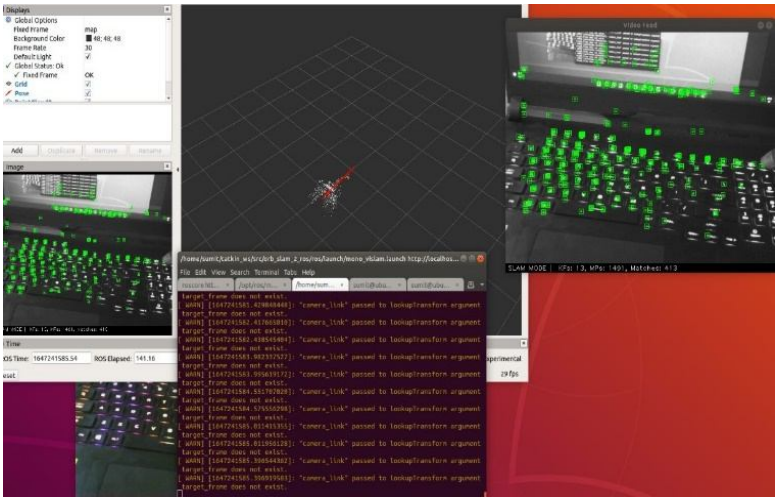


Fig. 2 – 3D Pose Estimation Graph

It is crucial to get the best possible estimation from the detected features when used practically as SLAM can develop large scale/dense reconstruction with massive data processing needs.

Generally, RANSAC is used to select valuable features by retrieving the inlier set. This process is randomized.

B. Tracking in Visual SLAM

If a new keyframe is found, add it to the list of keyframes and update the properties of the map points recognized by the new keyframe. To minimize the outliers in a mapPointSet, you need to check for legitimate map points in at least three important frames. New map points are formed by triangulating the ORB feature points of the current key frame and its associated key frames. Use matchFeatures to find a match for each unmatched feature point in the current keyframe with other unmatched feature points in the connected keyframe. The current keyframe pose, the connected keyframe pose, and the entire map are all adjusted by local bundle adjustment.

Fig. 3 – Object of Mapping

C. Loop Closing

The loop closure discovery phase, after being processed by the local mapping process, examines the current keyframe and attempts to detect and close the loop. EvaluateImageRetrieval is used to find loop candidates by querying the database for photos that are visually similar to the current keyframe. It is valid if the candidate keyframe is not connected to the last keyframe and three of its adjacent keyframes are loop candidates.

estimatesGeometricTransform3D is used to calculate the relative position between the loop candidate frame and the current key frame when a valid loop candidate is found. A 3-D similarity transformation held in an affine 3-D object is represented by the relative pose. Then add the relative pose loop connection and update mapPointSet and vSetKeyFrames.

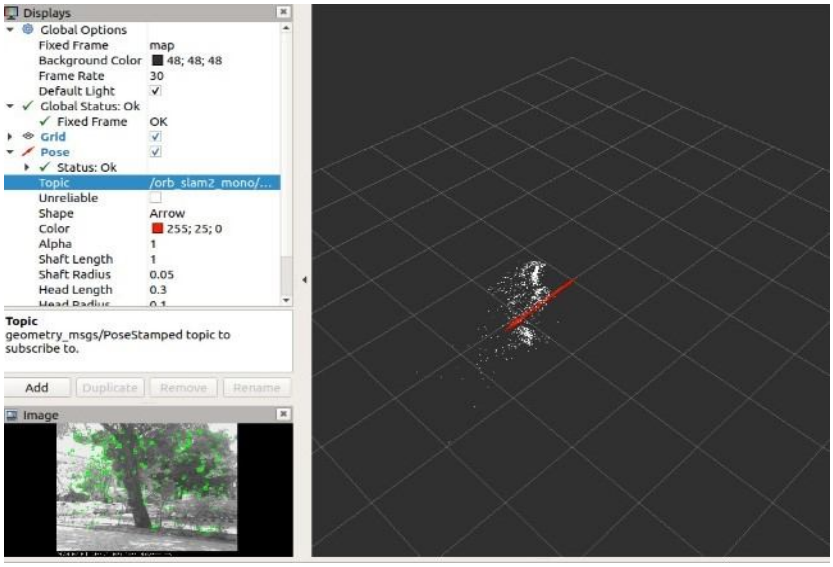


Fig. 4 – Point Map of the Object

D. Bundle Adjustment

Because mapping is not limited to frame rate, keyframe-based techniques estimate the map using only selected frames, called keyframes, allowing for more expensive but accurate bundle adjustment improvements.

This method additionally employs co-visibility information between keyframes and map points to determine which points to search for during tracking, allowing for large-scale operation. FAB-MAP is used to detect and delocalization loops. Pose graph optimization is used to close the loop, followed by structure-only bundle correction.

To ensure an optimal re-construction in the environs of the camera posture, the local mapping thread analyses new keyframes and performs local bundle adjustment.

III.ENCRYPTION

The length of the encryption key used to encrypt a data stream or file is called 256-bit encryption. Decrypting a 256-bit encrypted message requires 256 different combinations of hackers or crackers. This is almost impossible even on the most powerful computers. 256-bit encryption is typically used for data in transit or for data that travels over a network or internet connection. It is, nevertheless, used for sensitive and critical data, such as financial, military, and government-owned information. All sensitive and significant data must be protected using 192-bit or 256-bit encryption technologies, according to the US government.

The symmetric methodology has focused on employing substitution cypher to improve the traditional way of encryption. For cypher text, substitution techniques have been utilized the undeniable textual content is first translated into the corresponding ASCII code price of every alphabet on this symmetric algorithm. Both encryption and decryption are accomplished the use of the identical key. DES, RC2, RC4, IDEA, and other well-known symmetric key algorithms are examples. She delves into the details of many symmetric key algorithms before proposing a new one. Here you'll find both encryption and decryption algorithms.

A. Algorithm of Encryption

1) Step 1: Generate the letter's ASCII value.
2) Step 2: Generate the letter's binary value. [Binary value should be 8 digits, e.g., 00100000 for decimal 32 binary number]
3) Step 3: Reverse the binary number's 8 digits.
4) Step 4: As the Key, choose a four-digit divisor (>=1000).
5) Step 5: Subtract the divisor from the inversed number.
6) Step 6: Put the remainder in the first three digits and the quotient in the following five (the remainder and quotient can't be more than three and five digits long, respectively). If any of these are less than 3 or 5 digits, we must add the necessary amount of 0s (zeroes) to the left-hand side. As a result, this is the encrypted text.

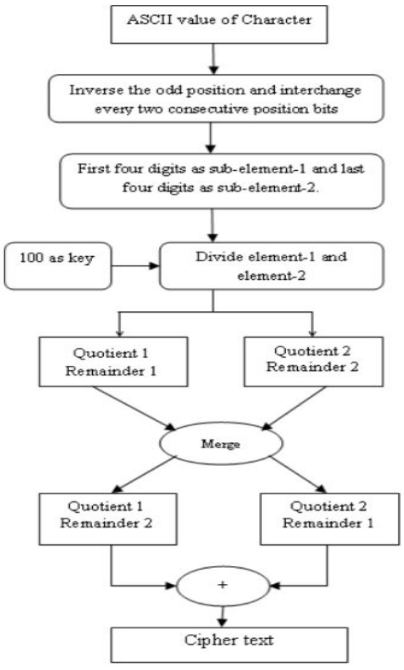


Fig. 5 – Architecture and Flow of Algorithm

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue V May 2022- Available at www.ijraset.com

B. Algorithm for Decryption

1) Step 1: Multiply the cypher text's last 5 digits by the Key.
2) Step 2: Multiply the first three digits of the encrypted text by the result obtained in the previous step.
3) Step 3: If the result from the previous step, step 2, is not an 8-bit number, we must convert it to one.
4) Step 4: Reverse the number to get the plain text, which is the original text.

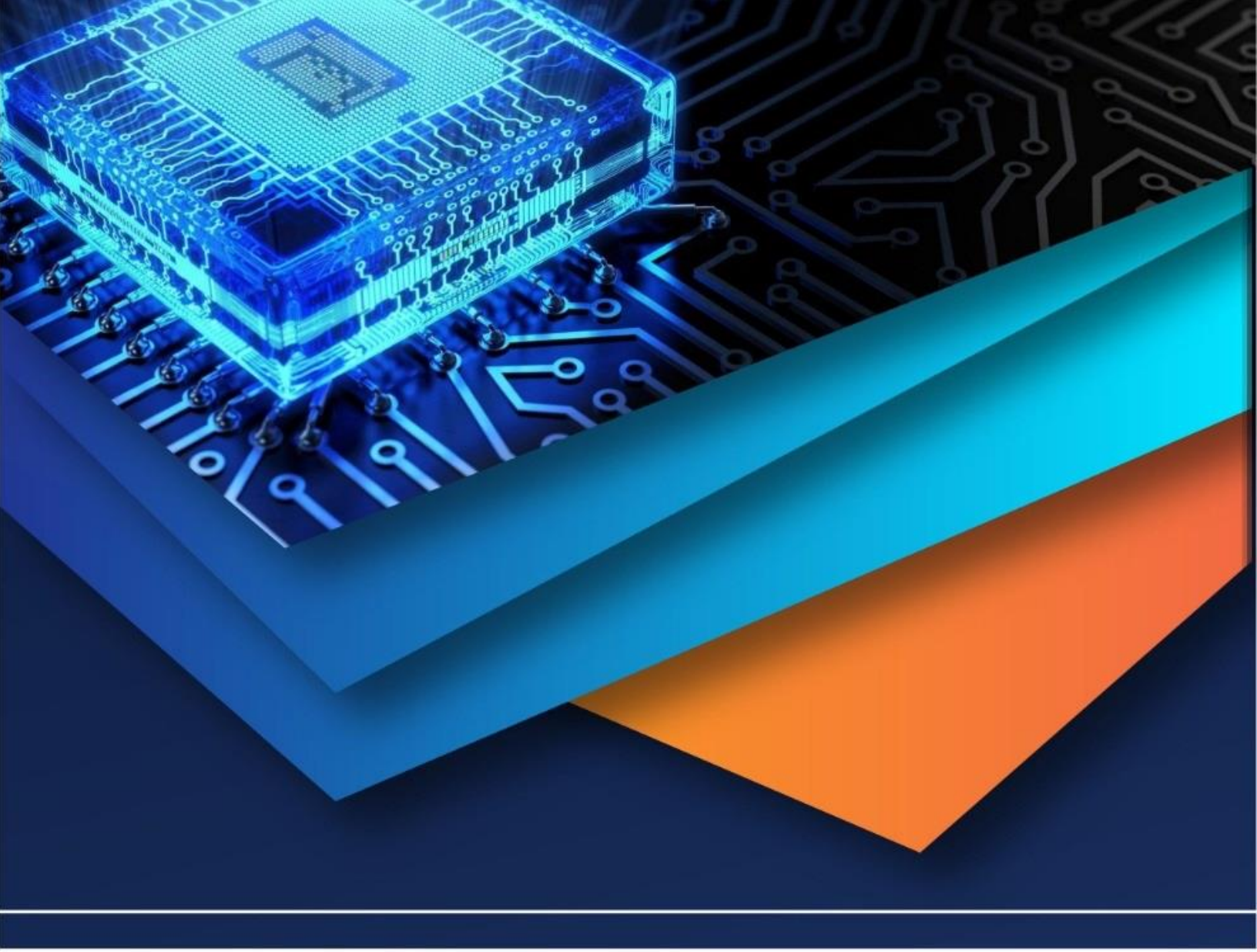
IV. CONCLUSIONS

This research work has allowed us to thoroughly study and understand the concepts of Visual Slam and Encryption. Whenever our UAV is in motion, its monocular Camera captures the feed and during the transmission of data from the aerial bot to the local machine, it stays encrypted and is virtually impossible to crack. All the methodology used were also supported by the base of ROS (Robotics Operating System) and Cyber Security. The future scope in the project will also lead to immense learning into various fields as well.

REFERENCES

[1] M. W. Mueller and R. D'Andrea, "Machine learning for high-speed corner detection," Int. J. Robot. Research, Dec. 2016
[2] S. Sun, X. Wang, Q. Chu, and C. d. Visser, Incremental nonlinear fault-tolerant control of a quadrotor with complete loss of two opposing rotors, 2020.
[3] Rosten and T. Drummond, "Machine learning for high-speed corner detection. European Conf. on Comp. Vis. (ECCV), 2006.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)