# Center of Gravity and Velocity Estimation of Unknown Orbiting Objects from Optical Flow

Muhammad Moughal

Soon-Jo Chung

Vincenzo Capuano

## Abstract

In spacecraft proximity operations and missions, such as on-orbit servicing, formation flying, and small bodies exploration, on-board vision based techniques are required for autonomous relative navigation of a chaser spacecraft with respect to a target object because ground-based communication is inefficient due to delays and lack of coverage. These techniques are grouped under the general problem of spacecraft pose estimation. Successful pose estimation provides the relative position and orientation of a target object with respect to the chaser spacecraft. In this project, we improved upon a portion of the monocular based pose estimation system for such targets with the idea that monocular based systems offer a solution with low mass, volume, and power consumption. We developed an Extended Kalman Filter based algorithm that utilizes an alternative derivation of optical flow to estimate the center of gravity and the relative velocity of a moving target with respect to a moving chaser. These values form a part of the overall pose of the target as required by the goal of spacecraft pose estimation.

## Introduction

Spacecraft pose estimation is the general problem of determining the position and orientation of a target object by a chaser spacecraft for missions requiring a close proximity approach and interaction between the chaser and target. These missions include orbital maneuvers, comet exploration, and more. To determine the pose of the target, the estimation of its related quantities such as the position of center of gravity (CG), translation velocity, etc., is required. In our system, this is accomplished through the design and implementation of the SEPS and CG filters. Together, these filters estimate the position and orientation of a moving target with respect to a moving spacecraft.

For this summer project, I focused on the CG EKF, i.e., the estimation of the related quantities to pose that are extractable from optical flow, such as the CG and relative velocity of the target. In the past weeks, I added on to the EKF simulations by adding Gaussian noise to mimic realistic conditions because it is necessary to make sure our EKF can filter out noise correctly. I also extended the functionality of the EKF by building the covariance parameters required for filtering the noises and utilizing least squares formulations to improve the overall accuracy of the EKF.

# Results

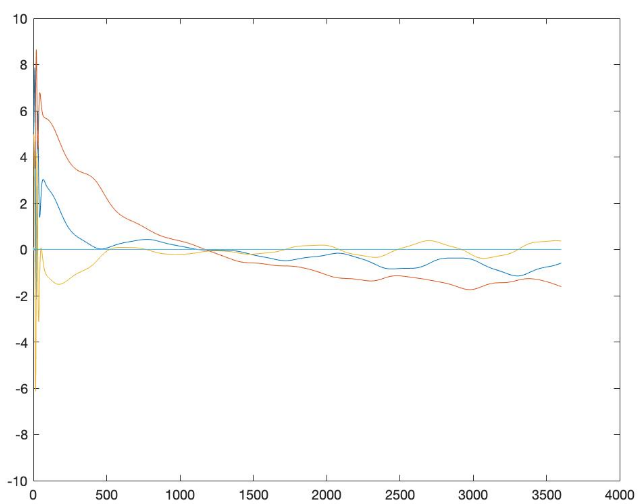


Figure 1: CG filter with a basic, identity covariance matrix

Above there are two example plots of CG from simulations with added Gaussian noises to values such as landmarks. These simulations run for 3600s with a time step of 1.00s, so there

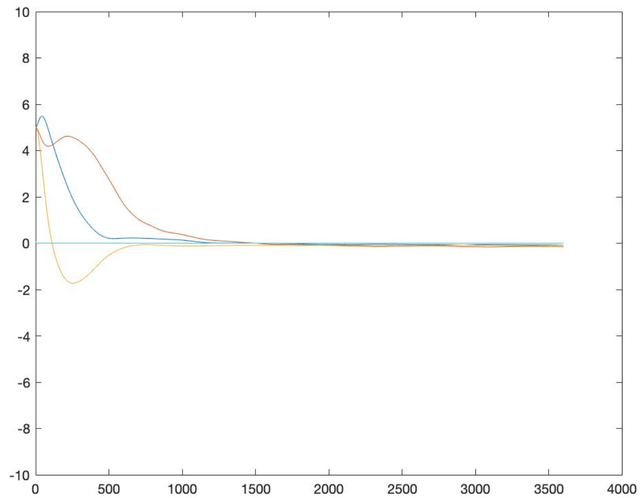


Figure 2: CG filter with a least squares based covariance matrix

are 3600 data points. Each data point represents the difference between the real CG value and the estimated CG value. The results show the difference between utilizing different covariance parameters. Figure 3 shows the results from using an identity covariance matrix and Figure 4 shows the result from using a covariance matrix developed from the least squares formulation. The results of CG estimation for both methods start with similar levels of errors, however the least squares method's error decreases more as time goes on. To get an estimate of improvement, we averaged the norm of the error vector of the state simulations with three simulations and found the decrease in error to be an average of ~60%.

# Conclusions

To summarize the past weeks' progress, we have developed and improved the optical flow based CG filter to estimate the CG and velocity of moving targets with respect to moving chasers. In combination with the SEPS filter, this CG filter determines the targets' overall poses. Some further steps now include improving the least squares methods for faster performance and better estimation of state. The next step of the CG filter's development is to modify the state vector of the filter to include the angular velocity of the target. This would provide us with a filtered estimate of the rotation as opposed to a measurement and as a result, the CG-SEPS system will determine the overall pose of the target more accurately. This would require the estimation of the target's inertial matrix and therefore modifying the state and measurement vectors of the filter. As time is limited, I will focus mostly on the former task which is to modify the least squares formulation for better overall performance of the current CG filter.

# Methods

## The Extended Kalman Filter (EKF)

The EKF in our pose estimation algorithm is used to estimate quantities such as the position of the center of gravity (CG) and translation velocity of the target. Together these values help us determine the pose (position and orientation) of the target. The filter's two main parameters include the state vector and the measurement vector. To get the values of the measurement vector, we use the optical flow equation to relate the 2D points on the image to the movement of the target. The regular optical flow equation $\dot{p} = v + w \times p$ is the derivative of the motion equation $p(t) = R(t)p + T(t)$. However, this equation assumes that only one of the objects in the chaser-target system is moving, i.e., going under both translation and rotation. Our chaser-target simulation is modeled after the case where the chaser is approaching the target within several meters and both objects are moving, so it takes the rotation and translation of both objects into consideration. The optical flow equation for this scenario is derived in Section 2.2 of (*1*) and is shown in Eq. (1):

$$\left.\frac{\partial o}{\partial t}\right|_{t=0} = \left.\frac{\partial P(p')}{\partial p'}\right|_{p'(0)} (\omega_c \times p' + \omega' \times p' - \omega' \times T'_0 - v') \tag{1}$$

, where $\omega_c$ and $p'$ are known.

In this new optical flow equation, the flow of the 2D image points, $o$, is related to the rotation of the camera, $w_c$, the rotation of the target, $w'$, the CG of the target, $T'_0$, and the relative velocity $v'$. Since $w_c$ is known, there are six unknowns: $w'$ and the combined quantity $w' \times T'_0 - v'$. Thus we can solve for these measurements using 3 points in a system of 6 linear equations. This will give us values for the vector $w' \times T'_0 - v'$ which is then utilized by the filter as explained below.

With this formulation in mind, we design the EKF's state vector as $\boldsymbol{x} = [\boldsymbol{r}', \boldsymbol{v}']$ and the measurement vector as $\boldsymbol{y} = [\boldsymbol{w}' \times \boldsymbol{r}' - \boldsymbol{v}']$. Here the $\boldsymbol{r}'$ vector is the same as the $T_0'$ vector mentioned earlier, i.e., the CG of the target. The filter estimates the values in the state vector. These values include the center of gravity $\boldsymbol{r}'$ and the translation velocity of the target $\boldsymbol{v}'$. The filter uses the values in the measurement vector and the prediction of state from the motion model to estimate the state vector. The values of the measurement vector are developed through the observations of the camera and the application of the optical flow as explained above. This process is illustrated in the architecture of the filter in Figure 2.

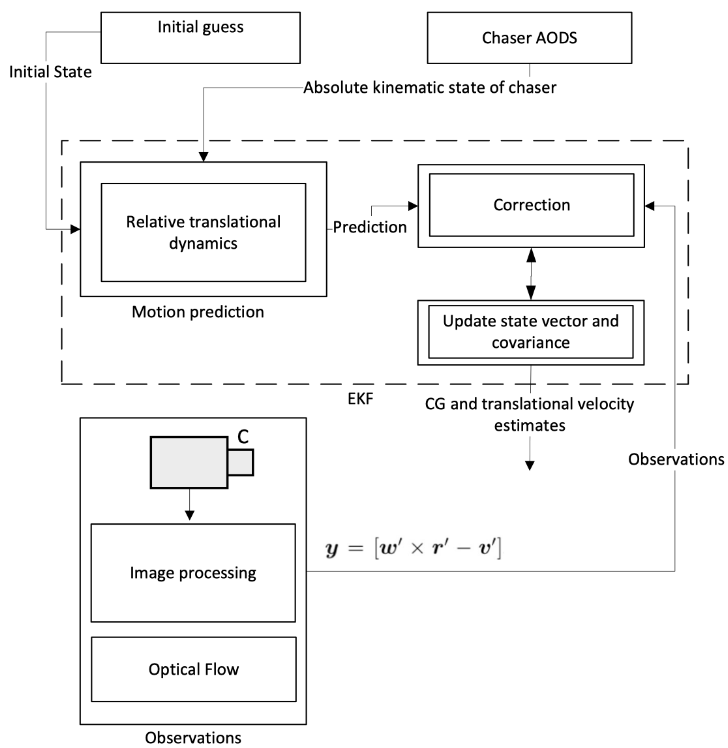


Figure 3: The architecture of the EKF

## Gaussian Noises

The gaussian noise generates values that are normally distributed which is required to mimic realistic conditions in our pose estimation simulation. In order to add noise to our measurement vector, a noise with a certain variance is added to values such as the landmark positions (the

3D points) of the target. The noise of values such as the landmark positions then propagates through the simulation and combines to result in a certain noise of the measurement vector y. We implement the gaussian noise through the *randn* function in Matlab. In order to add a noise with standard deviation *a* to a certain value *x*, we perform the following operation: $a * randn + x$. The variance of this noise is $a^2$. This variance is used to develop covariances as shown in the next section.

## Covariances

### Estimating Covariance of the Measurement

As mentioned above, by generating gaussian noises with a certain standard deviation and variance, our measurement vector y has a noise that the EKF then filters out for the updated state vector x. For this process, we develop covariance matrices of the measurement vector y. Recall from Section 2.2 of (*1*) that we can solve for values of the measurement vector y by setting up a system of equations with 3 points. Solving for y requires solving the system $Ay = b$. Note that this formulation of the problem is similar to the least squares problem. We can exploit this idea to then estimate the covariance of y by $(\boldsymbol{A'A})^{-1}\sigma^2$ where $\sigma^2$ is the variance of the noise of the measurement (*2*).

### The Variance of y

For the variance of y, the estimation of the measurement's covariance requires manually calculating variance of the noise. We can find this through calculating the variance of optical flow relation illustrated in Figure 1: $\sigma_y^2 = \sigma_{do/dt}^2 + \sigma_{wc}^2 * \sigma_{ps}^2$. Since we want to minimize error with filtering, we need to make sure that the larger covariances receive less weight compared to the smaller covariances. This is accomplished through implementing the square root of the information matrix $\boldsymbol{R_\Omega}$. Here, $\boldsymbol{R_\Omega} = \Omega^{-1}$, where $\Omega$ is the variance $\sigma_y^2$. From (*2*), $(Ax - b^T)\boldsymbol{R_\Omega}(Ax - b)$ simplifies to $\|\sqrt{\boldsymbol{R_\Omega}}(Ax - b)\|^2$ and thus we can find the covariance of y using the corrected variance with $(\boldsymbol{A'R_\Omega A})^{-1}\sigma^2$.

# References

1. V. Capuano, K. Kim, A. Harvard, and S.-J. Chung, “Monocular-Based Pose Determination of Uncooperative Space Objects,” Acta Astronautica, 2019.

2. S.A. Van De Geer, "Least Squares Estimation," 2005.

# Acknowledgements

I would like to thank Caltech-JPL scientists and engineers including Dr. Vincenzo Capuano and Alexei Harvard for closely mentoring me on the project, along with Dr. Soon-Jo Chung for providing the opportunity and resources for this project. Lastly, I would like to acknowledge the Rawlings Research Program at Cornell and the Summer Undergraduate Research Program at Caltech for providing access to this valuable learning opportunity and setting up the summer logistics.