# OV2SLAM on EuRoC MAV Datasets: a Study of Corner Detector Performance

Anthony Burghoffer, Jeremy Seyssaud, Baptiste Magnier

# OV$^2$SLAM on EuRoC MAV Datasets: a Study of Corner Detector Performance

Anthony Burghoffer[1,2], Jérémy Seyssaud[1] and Baptiste Magnier[2]
[1]*CEA/DES/ISEC/DPME/SEIP/LNPA*, site de Marcoule, France
[2]*EuroMov Digital Health in Motion, Univ Montpellier, IMT Mines Ales,* Ales, France
anthony.burghoffer@cea.fr, jeremy.seyssaud@cea.fr, baptiste.magnier@mines-ales.fr

***Abstract*—Theoretically, SLAM (Simultaneous Localization and Mapping) systems acquire information from its environment with sensors, extract landmarks from the received data and estimate its location on a map based on the sensor measurements. EuRoC datasets is a batch of visual-inertial measurements from embedded stereo camera and inertia measurement unit in a Micro Aerial Vehicle (MAV). The MAV flights include eleven itineraries and took place in indoor environments: an industrial environment and two motion capture rooms. OV$^2$SLAM (Online and Versatile Visual SLAM) is an open-source visual feature points-based SLAM methods that is remarkably efficient. A feature points-based method extracts and tracks keypoints because they represent stable features. The native keypoint detection method in OV$^2$SLAM tested in the EuRoC MAV datasets is a well-known KLT (Kanade-Lucas-Tomasi) corner detector. Nevertheless, no other detector was experimented on this SLAM method. This paper enables the investigation of which corner detector is optimum for OV$^2$SLAM method on the EuRoC MAV datasets. Overall, the experiments are led on 10 itineraries containing 28 058 stereo-pair images in all. Thus, by varying the parameter of the Gaussian influencing the detection of the keypoints, a global score based on different statistics is calculated in relation to the ground truth to classify which pair detector/parameter is optimal on these datasets.**

***Index Terms*—OV$^2$SLAM, EuRoC datasets, Corner detection**

## I. Introduction, Context and Motivation

*1) SLAM (Simultaneous Localisation And Mapping):* SLAM is a momentum technological method suitable for autonomous systems such as robotics or self-driving cars. It allows a system to analyze and describe an unknown environment, which could be unsafe, make a map out of it and place itself on it.

In 1986, the first statement about the SLAM problem was made [18]. Thereafter, the pioneer work describing the structure of the SLAM problem was led in 2006 [3]. Since then, techniques and sensors have evolved to become more and more efficient. Essentially, the SLAM can split into two parts: (*i*) the algorithms and (*ii*) the used sensors. The software layer of a SLAM estimates both: the location of the system and environmental mapped landmarks. These two inputs are estimated, so the truth positions of the system and the landmarks are never known. Therefore, the algorithms however compelling rely on probabilistic equations and are a continuous active subject for scientific inquiry, see the following references: [1], [4], [12], [19].

Consider a mobile vehicle equipped with a sensor and moving through an environment taking relative observations of a number of unknown landmarks, as diagrammed in Fig. 1. At a time instant $k$, the following quantities are defined:

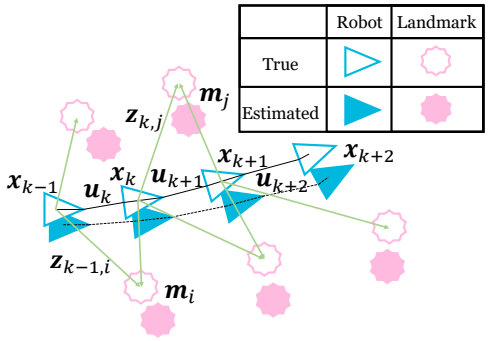


Fig. 1. Summary of the essential SLAM problem: a simultaneous estimation of both vehicle (or robot) and landmark locations is required. The true locations are never known or measured directly, whereas observations are made between true robot and landmark locations.

- $\mathbf{x}_k$: the state vector describing the location and orientation of the vehicle,
- $\mathbf{u}_k$: the control vector, applied at time $k$-1 to drive the vehicle to a state $\mathbf{x}_k$,
- $\mathbf{m}_i$: a vector describing the location of the $i$th landmark whose true location is assumed time invariant,
- $\mathbf{z}_{ik}$: an observation taken from the vehicle of the location of the $i$th landmark at time $k$; when there are multiple landmark observations at any one time or when the specific landmark is not relevant to the discussion, the observation will be written simply as $\mathbf{z}_k$.

In addition, the following sets are also defined:

- $\mathbf{X}_{0:k} = \{\mathbf{x}_0, \mathbf{x}_1, ..., \mathbf{x}_k\} = \{\mathbf{X}_{0:k-1}, \mathbf{x}_k\}$: the history of vehicle locations,
- $\mathbf{U}_{0:k} = \{\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_k\} = \mathbf{U}_{0:k-1}, \mathbf{u}_k$: the history of control inputs,
- $\mathbf{m} = \{\mathbf{m}_1, \mathbf{m}_2, ..., \mathbf{m}_n\}$: the set of all landmarks,
- $\mathbf{Z}_{0:k} = \{\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_k\} = \{\mathbf{Z}_{0:k-1}, \mathbf{z}_k\}$: the set of all landmark observations.

In probabilistic form, the SLAM problem requires that the probability distribution is computed for all times $k$. Given the recorded observations and control inputs up to and including time $k$ together with the initial state of the vehicle, this probability distribution describes the joint posterior density of the landmark locations and vehicle state at each time $k$ and is denoted $P(\mathbf{x}_k, \mathbf{m}|\mathbf{Z}_{0:k}, \mathbf{U}_{0:k}, \mathbf{x}_0)$.

(a) Left image (b) Right image (c) Left image (d) Right image

Fig. 2. Selected images of the embedded videos [2], images of 752×480 pixel size. Images in (a)-(b) are extracted from the first flight in the Machine Hall (MH_01) whereas in (c)-(d) from the first path in the Vicon Room 1 (V1_01).

TABLE I
BRIEF DESCRIPTION OF THE EUROC DATA-SETS

| Name | Description |
|---|---|
| MH_01_easy | good texture, bright screen |
| MH_02_easy | good texture, bright screen |
| MH_03_medium | fast motion, bright screen |
| MH_04_difficult | fast motion, dark screen |
| MH_05_difficult | fast motion, dark screen |
| V1_01_easy | slow motion, bright screen |
| V1_02_medium | fast motion, bright screen |
| V1_03_difficult | fast motion, motion blur |
| V2_01_easy | slow motion, bright screen |
| V2_02_medium | fast motion, bright screen |
| V2_03_difficult | fast motion, motion blur |

*2) EuRoc MAV datasets:* The EuRoC MAV datasets [2] are visual-inertial datasets collected on-board a Micro Aerial Vehicle (MAV). They include especially the embedded videos (stereo-pair images, see Fig.2), the inertial measurement unit (IMU) of eleven flights and their millimeter accurate position ground truth from a laser tracking system. Five datasets takes place in an industrial environment called Machines Room. The six remaining flights are in a custom three-dimensional (3D) environment in a room called Vicon Room equipped with motion capture cameras (100 Hz). The datasets contain six-dimensional (6D) pose ground truth and a detailed 3D scan of the environment. The provided datasets are ranged from slow flights under good visual conditions to dynamic flights with motion blur and poor illumination, enabling researchers to thoroughly test and evaluate their algorithms; the perturbations are reported in Tab.I.

*3) OV²SLAM:* Online and Versatile Visual SLAM (OV²SLAM) is an open-source visual SLAM design for real-time applications [4]. The overall concept integrates a succession of processes which are parallelized for each image (*i.e.*, image 2 processing begins during image 1 processing). The simplified structure of OV²SLAM is diagrammed in Fig. 3. The **Image pre-processing** stage receives new images and applies a contrast enhancement. The **Key-points tracking** is a feature tracker using a well-known corner detector denoted KLT [16] (see Tab.II); the corner detection is discussed in the following sections. The **Outlier filtering** step allows outliers removing that can still occur in the tracking process and a RANSAC filtering is applied based on the epipolar constraint. Thanks to this filtering step, no outliers are left before estimating the camera position. The **Pose estimation** stage uses a robust Huber cost function to minimize the 3D key-points re-projection errors. Finally, the **Keyframe creation** stage triggers keyframe related functions such mapping thread or the loop closer thread.

In this study, only the **Key-points tracking** part of the OV²SLAM is fitted in order to improve the SLAM on the EuRoc MAV datasets.

## II. CORNER DETECTION: IMPORTANT FEATURES TO TRACK

### A. Keypoint detection

There are several methods to perform a visual SLAM: the feature-based methods, the direct methods and the RGB-Depth methods. OV²SLAM is a feature points-based method: it extracts feature points from an image and keeps track of those keypoints in a 3D space. In digital images, corners are important features which are visually distinguishable because they are localized at the maxima of edges absolute curvature (detailed in [20]). Corner points represent anchor points, these features should be consistently identified from an image and the more the features are extracted reliably, the more your SLAM will be accurate.

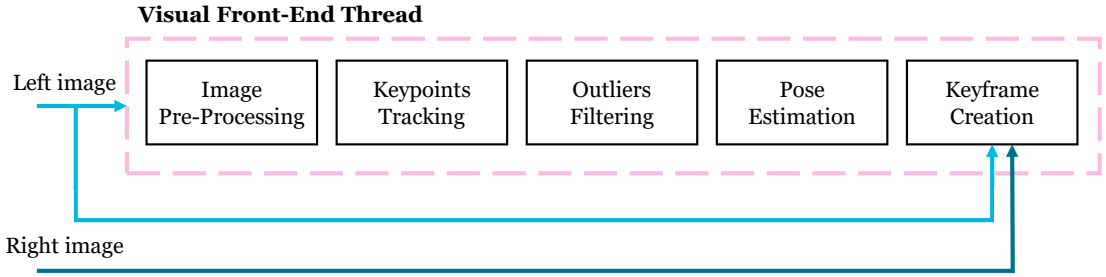


Fig. 3. Architecture of OV²SLAM with real-time processing presented in [4]. The front-end thread figured contain the corner detection methods. It is responsible for estimating the pose of the camera in real-time.

TABLE II

CORNERNESS MEASURE FORMULAS COMPUTED FROM IMAGE DERIVATIVES. HERE, $\overline{\bullet}$ DENOTES THE CONVOLUTIONS WITH A GAUSSIAN WITH A STANDARD DEVIATION $\sigma > 0$ OF IMAGES DERIVATIVES $I_x$, $I_y$ AND THEIR INNER PRODUCTS. AS A REMINDER, $(\lambda_1, \lambda_2)$ REPRESENTS THE EIGENVALUES OF THE STRUCTURE TENSOR $\mathcal{M}$ IN EQ. (1).

| Name | Cornerness Measure Formula | Parameter(s) | Reference |
|---|---|---|---|
| KLT | $Min\left(\lambda_1, \lambda_2\right) = Min\left(\overline{I_x}^2 + \overline{I_y}^2 \pm \sqrt{\overline{I_x}^2 + \overline{I_y}^2 - 4 \cdot \overline{I_x I_y}}\right)$ | $\sigma$ | [16] |
| F | $\frac{Det(\mathcal{M})}{Trace(\mathcal{M})} = \frac{\lambda_1\lambda_2}{\lambda_1+\lambda_2} = \frac{\lambda_1\lambda_2}{\lvert\nabla\overline{I}\rvert^2} = \frac{\overline{I_x}^2\overline{I_y}^2 - \overline{I_x I_y}^2}{\overline{I_x}^2 + \overline{I_y}^2}$ | $\sigma$ | [6] |
| HS | $Det(\mathcal{M}) - k \cdot (Trace(\mathcal{M}))^2 = \overline{I_x}^2\overline{I_y}^2 - \overline{I_x I_y}^2 - k \cdot \left(\overline{I_x}^2 + \overline{I_y}^2\right)^2$ | $\sigma,\ k$ | [9] |
| Ro | $Det(\mathcal{M}) = \lambda_1\lambda_2 = \overline{I_x}^2\overline{I_y}^2 - \overline{I_x I_y}^2$ | $\sigma$ | [14] |
| KZ | $\frac{1}{\left(\lambda_1^{-p} + \lambda_2^{-p}\right)^{1/p}}$ | $\sigma,\ p > 0$ | [10] |

### B. Reliable corner detectors for repeatability

The importance and interest in corner detection lie notably in its application in image matching, tracking, motion estimation, panoramic stitching, object recognition, and 3D reconstruction [15]. Repeatability is the main evaluation metric widely used for interest point matching, where the obtained points must be independent of varying image conditions [15] [13]. From [20], the definition of "Repeatability" is given by: *Given two images of the same object or scene, taken under different viewing conditions, a high percentage of the features detected on the scene part visible in both images should be found in both images.* The repeatability of the keypoint detector allows the improvement of the SLAM. Hence, an investigation of several corner detectors regarding the repeatability is led in [17] regarding real sequences of [5] and the most reliable techniques compute the structure tensor $\mathcal{M}$. Considering a gray-level image $I$ and its partial derivatives: $I_x$ and $I_y$, the 1st image derivative along the $x$ and $y$ axis respectively, and, $I_{xy}$, the crossing derivative of $I$, the structure tensor $\mathcal{M}$ is given by:

$$\mathcal{M} = \begin{pmatrix} \overline{I_x^2} & \overline{I_x I_y} \\ \overline{I_x I_y} & \overline{I_y^2} \end{pmatrix}, \tag{1}$$

where $\overline{\bullet}$ indicates convolution with a Gaussian filter $G$ of standard deviation $\sigma$: $G(\sigma, x, y) = \frac{1}{2\pi\sigma^2} \cdot \mathrm{e}^{-\frac{x^2+y^2}{2\sigma^2}}$ with $\sigma \in \mathbb{R}_+^*$ and $(x, y) \in \mathbb{R}^2$.

Spatial averaging, here tied to $\sigma$ parameter, distributes this information over a neighborhood. The eigenvectors of the gradient structure tensor indicate local orientation, whereas eigenvalues $(\lambda_1, \lambda_2)$ give the strength or magnitude as a measure: (*i*) in flat regions they are negligible, (*ii*) in the edges $\lambda_1$ or $\lambda_2$ is small depending on the horizontal or vertical edge, and (*iii*) noticeably both values $\lambda_1$ and $\lambda_2$ are large in corner points –the eigenvalue representations are discussed in [7], part III-B–. Based on this assumption, various corner measurement formulations have been proposed; they are listed and denominated in the Tab. II and summarized in [17]. Note that KZ is equivalent modulo for the choice of a suitable matrix norm and a normalization constant to: (*i*) F when $p = 1$, (*ii*) KLT when $p \to +\infty$, and, (*iii*) $\sqrt[p]{2}$R for $p \to 0$. In this study, $p$ is fixed to 2.

In a nutshell, these five corner detection techniques have in common the tensor $\mathcal{M}$ of eq. (1), which is tied to the same low-pass filter parameter, here denoted $\sigma$: the standard deviation of the Gaussian. These corner detection techniques are frequently utilized relating to their real-time application, as in [8]. The objective here is to determine which corner detector with which optimum parameter $\sigma$ allows to create a reliable SLAM on the EuRoC MAV datasets.

## III. EVALUATION AND RESULTS

The objective of this paper is to observe how the modification of the corner detector or/and its parameter affects the performance of the OV$^2$SLAM algorithm on the EuRoC MAV Datasets. The choice of a $\sigma$ parameter value tied to the Gaussian $G$ for the corner detectors is usually made empirically because a too large value can delocalize the key-point position and will "disrupt" the repeatability, as experimented in [17]. Indeed, when the $\sigma$ value increases, the key-points can get misplaced increasingly (see example in [13]); additionally, some of them could be merged. Meanwhile, too small $\sigma$ values may limit the detection of structures and will result the low repeatability ratio for matching. Therefore, for each detector, the $\sigma$ value is studied here for each corner detector presented in the Tab.II. In the native OV$^2$SLAM algorithm, the corner detector is the KLT method.

### A. Absolute pose error

The study will focus on the notion of absolute errors of positioning the drone in space. The absolute pose error (*ape*) is calculated as follows:

$$ape = \sqrt{x_e^2 + y_e^2 + z_e^2}. \tag{2}$$

Examples of three *ape* along the first Machine Hall itinerary are shown in Fig. 4. Thereafter, a maximum, a minimum, a median, a mean or a standard deviation error of *ape* per itinerary can be computed, as detailed in the next section.

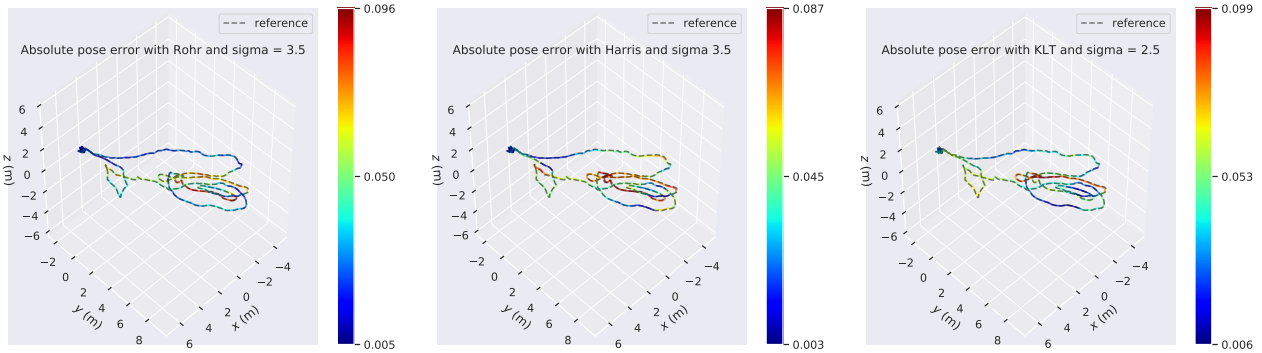


Fig. 4. 3D representation of the best path estimated by OV$^2$SLAM with Rohr $\sigma = 3,5$ (left), Harris $\sigma = 3,5$ (center) and KLT $\sigma = 2,5$ (right). The represented path is tied to the Machine Hall 01 video containing 3 682 stereo-pair images in all.

### *B. Evaluation*

Ten out of eleven of the itineraries proposed by EuRoc will be analyzed, corresponding to 28 058 stereo-pair images in all. The eleventh itinerary led the algorithm to failure with the worst performing $\sigma$ parameter/detector pairs. The five corner detectors presented in Tab. II with a parameter $\sigma$ between 0.5 and 4.5 are tested on each flight (with a step of 0.5). Each combination will be compared to the ground truth. Hence, the following errors are computed:

- $M_e$: the maximum error,
- $m_e$: the minimum error,
- $\overline{M}_e$: the mean error,
- $Me_e$: the median error,
- $STD_e$: the standard deviation error.

These statistic errors are extracted from the comparison for each detector and each parameter. Thereafter, each extracted element is averaged over the ten runs and normalized as a function of the worse error for each entity among all the detectors and all the parameters. In order to rank the couples detector/$\sigma$ parameter, a final Score is computed by:

$$\text{Score} = \frac{1}{5} \cdot \left[\mathcal{N}(M_e)+\mathcal{N}(m_e)+\mathcal{N}\left(\overline{M}_e\right)+\mathcal{N}(Me_e)+\mathcal{N}(STD_e)\right], \tag{3}$$

where $\mathcal{N}$ represents the normalization function.

The 5 performance indicators in millimeters (mm) and total scores are reported in Tab. III. Usually, the best performing detector on the EuRoC MAV datasets is the Rohr detector with $\sigma$=3,5. Nevertheless, the lower ranked detector/$\sigma$ pairs at the top of the leaderboard do not have a large score gap. The first six detectors can be considered equivalent. It is noteworthy that the Rohr detector appears three times in the top 6, and usually performs better than other detectors. Another interesting point is that the well-known Harris detector is mostly found in the middle and bottom of the tab. Also, Kenney is not reliable on this dataset. KLT detector is homogeneously repaired in the tab. however it has mostly a low mean error. Forstner has results rather grouped between the top end the middle of the tab.; it is less affected by $\sigma$ than the other detectors.

As a visual result, the Fig. 4 presents the estimated path taken by the drone during the flight in the Machine Hall and the ground truth. In the Harris and KLT path, the presence of error is more spread out than in the Rohr path. In addition, the maximal *ape* location in the itineraries is different for the various detectors. In average, the Rohr detector has a lower ape during the flight but still as a maximum *ape* equivalent to the other two detectors.

TABLE III

RANKING OF DETECTOR/$\sigma$ PAIRS. FOR EACH PAIR, THEIR AVERAGE PERFORMANCES ON THE TEN SEQUENCES OF THE EUROC MAV DATASET AND THEIR RESULTING SCORES ARE FIGURED. THE BOLDED SCORES CORRESPOND TO THE BEST SCORE FOR EACH COLUMN. $M_e$ STANDS FOR MAXIMUM ERROR, $m_e$ FOR MINIMUM ERROR, $\overline{M}_e$ FOR MEAN ERROR, $Me_e$ FOR MEDIAN AND $STD_e$ FOR STANDARD DEVIATION ERROR IN MILLIMETERS (MM).

| Rank | Detector | $\sigma$ | $M_e$ | $\overline{M}_e$ | $Me_e$ | $m_e$ | $STD_e$ | Score |
|---|---|---|---|---|---|---|---|---|
| 1 | Rohr (Ro) | 3,5 | **156,23** | 68,43 | 64,98 | 9,49 | **27,73** | **0,464** |
| 2 | Rohr (Ro) | 0,5 | **162,40** | **65,74** | **60,98** | 10,15 | **28,56** | **0,468** |
| 3 | KLT | 2,5 | 166,78 | **65,20** | **60,74** | 10,10 | **29,14** | **0,470** |
| 4 | Forstner (F) | 0,5 | 177,60 | 68,92 | 64,21 | **7,35** | 32,96 | **0,471** |
| 5 | Forstner (F) | 2 | 165,45 | 69,46 | 63,54 | 9,01 | 29,90 | **0,471** |
| 6 | Rohr (Ro) | 1,5 | 175,78 | 70,24 | 63,43 | **8,39** | 33,33 | 0,480 |
| 7 | Harris (HS) | 3,5 | 170,23 | **66,33** | 63,12 | 10,99 | **29,12** | 0,487 |
| 8 | KLT | 1,5 | 166,02 | **66,22** | **62,47** | 11,84 | 27,81 | 0,488 |
| 9 | Harris (HS) | 2,5 | 166,10 | 70,22 | 64,19 | 10,23 | 31,54 | 0,491 |
| 10 | Harris (HS) | 1 | 177,91 | 69,14 | 63,49 | 10,76 | 29,83 | 0,495 |
| 11 | Forstner (F) | 1,5 | **155,50** | 72,01 | 69,85 | 11,03 | **29,04** | 0,496 |
| 12 | Forstner (F) | 2,5 | 168,45 | 70,82 | 66,51 | 11,15 | 30,53 | 0,503 |
| 13 | Kenney (KZ) | 1,5 | 181,95 | 74,78 | 69,12 | **8,36** | 34,76 | 0,504 |
| 14 | Harris (HS) | 1,5 | 179,52 | 72,14 | 67,11 | 9,82 | 33,27 | 0,507 |
| 15 | Kenney (KZ) | 1 | 185,76 | 72,14 | 65,18 | 10,32 | 32,32 | 0,510 |
| 16 | Rohr (Ro) | 3 | 209,15 | 70,99 | 65,43 | **8,75** | 34,25 | 0,512 |
| 17 | Kenney (KZ) | 2 | 170,72 | 73,23 | 69,18 | 10,10 | 34,81 | 0,514 |
| 18 | Rohr (Ro) | 4,5 | **157,76** | 73,87 | 69,86 | 11,24 | 33,18 | 0,514 |
| 19 | KLT | 0,5 | 171,82 | **65,79** | **60,41** | 14,43 | 30,64 | 0,523 |
| 20 | Forstner (F) | 3 | 164,54 | 69,81 | 66,52 | 13,21 | 31,57 | 0,523 |
| 21 | KLT | 2 | **159,76** | 70,36 | 63,80 | 13,96 | 31,51 | 0,525 |
| 22 | Harris (HS) | 2 | 204,14 | 69,63 | 64,64 | 10,31 | 35,57 | 0,526 |
| 23 | Forstner (F) | 1 | 172,60 | 71,82 | 65,07 | 12,32 | 34,11 | 0,527 |
| 24 | KLT | 1 | 188,48 | 75,61 | 70,01 | 10,50 | 35,12 | 0,534 |
| 25 | KLT | 3,5 | 191,64 | 74,72 | 69,70 | 10,85 | 35,36 | 0,538 |
| 26 | Forstner (F) | 4,5 | 188,00 | 76,44 | 71,79 | 11,64 | 33,84 | 0,545 |
| 27 | Rohr (Ro) | 2,5 | 172,27 | 74,31 | 72,59 | 13,92 | 31,33 | 0,549 |
| 28 | KLT | 4,5 | 194,78 | 84,03 | 76,29 | **8,76** | 38,81 | 0,552 |
| 29 | Forstner (F) | 3,5 | 200,51 | 76,37 | 69,15 | 14,10 | 32,43 | 0,569 |
| 30 | KLT | 4 | 209,17 | 79,14 | 70,85 | 11,98 | 37,25 | 0,574 |
| 31 | Kenney (KZ) | 4 | 186,21 | 81,20 | 76,64 | 13,56 | 36,42 | 0,585 |
| 32 | KLT | 3 | 211,70 | 75,52 | 70,06 | 12,85 | 39,83 | 0,586 |
| 33 | Kenney (KZ) | 3 | 238,07 | 74,26 | 69,74 | 14,09 | 35,23 | 0,598 |
| 34 | Kenney (KZ) | 0,5 | 237,28 | 74,26 | **62,02** | 13,49 | 41,51 | 0,599 |
| 35 | Harris (HS) | 3 | 261,85 | 78,82 | 73,16 | 12,60 | 36,49 | 0,612 |
| 36 | Kenney (KZ) | 4,5 | 192,44 | 84,04 | 80,70 | 15,26 | 35,43 | 0,613 |
| 37 | Rohr (Ro) | 2 | 188,46 | 87,06 | 79,55 | 16,83 | 36,88 | 0,634 |
| 38 | Kenney (KZ) | 2,5 | 217,77 | 85,00 | 80,43 | 15,72 | 39,38 | 0,646 |
| 39 | Rohr (Ro) | 4 | 289,82 | 85,06 | 78,96 | 10,82 | 45,83 | 0,656 |
| 40 | Harris (HS) | 0,5 | 258,39 | 79,63 | 70,65 | 15,74 | 43,62 | 0,661 |
| 41 | Forstner (F) | 4 | 274,40 | 91,08 | 79,89 | 11,27 | 47,09 | 0,665 |
| 42 | Rohr (Ro) | 1 | 284,10 | 81,03 | 64,61 | 14,29 | 56,12 | 0,694 |
| 43 | Kenney (KZ) | 3,5 | 320,53 | 91,74 | 75,44 | 11,56 | 65,26 | 0,746 |
| 44 | Harris (HS) | 4,5 | 337,12 | 105,91 | 90,96 | 19,53 | 57,50 | 0,856 |
| 45 | Harris (HS) | 4 | 312,13 | 141,23 | 140,54 | 19,38 | 61,92 | 0,973 |

## IV. Conclusion and Discussion

Corners represent stable features possessing the defined characteristics of a robust point of interest, they remain an active research field for machine vision researchers like SLAM methods. Structure tensor-based approaches for corner detection are currently utilized for their extraction, as well-known Harris or KLT methods, but not only. In this work, we have achieved a study of the impact of the choice of the corner detector and its Gaussian parameter to the performance of a visual SLAM. The tools of this study were the OV$^2$SLAM [4] method applied on the EuRoC MAV datasets [2], both made available for research and improvement of SLAM systems. In that respect, OV$^2$SLAM uses corner detection which is the center of this study. Five corner detectors having reliable repeatability were selected from a prior work [17], namely: Harris (H), Rohr (Ro), Kenney (KZ), Forstner (F) and Kanade-Lucas-Tomasi (KLT). Five performance indicators have been extracted from the passage of the different detector/$\sigma$ pairs on the EuRoC itineraries: namely the maximum, minimum, mean, median and Standard deviation errors. These indicators allowed to determine the best scores among forty-five detector/$\sigma$ pairs. This methodology led to the following conclusion: usually Rohr detector is a better performing detector on the ten first itineraries proposed by EuRoC datasets.

More experimentation on other and different datasets with various contexts is needed to find if the Rohr detector generally performs better than other techniques. Indeed, we must remain critical of these results as they are only valid for the itineraries provided by the EuRoC MAV datasets.

An accumulation of uncertainty in the position measurement can lead to the SLAM system stalling. It is possible to make up for a drift in the position estimate with corrective solutions. These methods are called "Loop Closure". Usually, they involve threads running in parallel with front-end processing. Technically, the system looks for matches between newly detected interest points and previously recorded keypoints. These different methods are based on various types of technology. In [21], a visual sequence-based loop-closure detection pipeline is proposed. In [22], visual SLAM methods based on deep learning are reviewed. In both approaches, the performances of visual SLAMs are greatly improved. It is therefore interesting to carry out continuous improvement on the detection of corners (or other types of interest points) in conjunction with loop closure methods.

## References


[1] G. Bresson, Z., Alsayed, L., Yu, and S. Glaser. Simultaneous localization and mapping: A survey of current trends in autonomous driving. IEEE Trans. on Intelligent Vehicles, vol. 2, n°3, pp. 194–220, 2017.

[2] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M.W. Achtelik, and R. Siegwart. "The EuRoC micro aerial vehicle datasets." Int. J. of Robotics Research, vol. 35, n°10, pp. 1157–1163, 2016.

[3] H. Durrant-Whyte and T. Bailey. "Simultaneous localization and mapping: part I." IEEE Robotics and Automation Magazine, vol. 13, n°2, pp. 99–110, 2006.

[4] M. Ferrera, A. Eudes, J. Moras, M. Sanfourche and G. Le Besnerais. "OV$^2$ SLAM: A fully online and versatile visual SLAM for real-time applications." IEEE Robotics and Automation Letters, vol. 6, n°2, pp. 1399–1406, 2021.

[5] M. Ferrera, V. Creuze, J. Moras and P. Trouvé-Peloux, "AQUALOC: An Underwater Dataset for Visual-Inertial-Pressure Localization." Int. Journal of Robotics Research, vol.38, n°14, pp. 1549–1559, 2019.

[6] W. Forstner, and E. Gulch, "A fast operator for detection and precise location of distinct points, corners and circular features." Fast Processing of Photogrammetric Data, pp. 281–305, 1987.

[7] K. Grauman and B. Leibe, "Visual object recognition." Synthesis lectures on artificial intelligence and machine learning, vol. 5, n°2, pp.1–181, 2011.

[8] O.Haggui, C. Tadonki, L. Lacassagne, F. Sayadi and O. Bouraoui, "Harris corner detection on a NUMA manycore." Future Generation Computer Systems. Elsevier, vol. 88, pp. 442–452, 2018.

[9] C. G. Harris and M. J. Stephens, "A Combined Corner and Edge Detector." Alvey Vision Conference, pp. 147–151. 1988.

[10] C. S. Kenney, M. Zuliani, M. and B. S. Manjunath, "An axiomatic approach to corner detection." IEEE CVPR, Vol. 1, pp. 191–197. 2005.

[11] F. Mokhtarian and F. Mohanna, "Performance evaluation of corner detectors using consistency and accuracy measures." CVIU, vol. 102, n°1, pp. 81–94, 2006.

[12] J. A. Placed, J. Strader, H. Carrillo, et al. "Survey on active simultaneous localization and mapping: State of the art and new frontiers." IEEE Transactions on Robotics, 2023.

[13] V. Rodehorst and A. Koschan, "Comparison and evaluation of feature point detectors." International Symposium Turkish-German Joint Geodetic Days, 2006.

[14] K. Rohr, "Localization properties of direct corner detectors." Mathematical Imaging and Vision, vol. 4, pp. 139—150, 1994.

[15] C. Schmid, R. Mohr and C. Bauckhange, "Evaluation of Interest Point Detectors." IJCV, vol. 37, pp. 151–172, 2010.

[16] J. Shi and C. Tomasi, Good features to track. IEEE CVPR, pp.593–600, 1994.

[17] G. S. Shokouh, B. Magnier, B., Xu, and P. Montesinos, "Repeatability Evaluation of Keypoint Detection Techniques in Tracking Underwater Video Frames." Workshop on Computer Vision for Analysis of Underwater Imagery, 2022.

[18] R. C. Smith and P. Cheeseman, "On the representation and estimation of spatial uncertainty." The International J. of Robotics Research, vol. 5, n°4, pp.56–68, 1986.

[19] T. Jin, W. Tang and W. Wang, "Robust Visual SLAM systems with Constrained Sub-pixel Corners and High-quality Line Features." ICCV, pp. 101–106, 2022.

[20] T. Tuytelaars and K. Mikolajczyk, "Local Invariant Feature Detectors: A Survey." Foundations and Trends® in computer graphics and vision, vol. 3, n°3, pp. 177–280, 2008.

[21] K.A. Tsintotas, L. Bampis and A. Gasteratos, "The revisiting problem in simultaneous localization and mapping: A survey on visual loop closure detection." IEEE T-ITS, vol. 23, n°11, pp. 19929–19953, 2022

[22] K.A. Tsintotas, L. Bampis, S. An, G.F. Fragulis, S.G. Mouroutsos and A. Gasteratos, "Sequence-based mapping for probabilistic visual loop-closure detection." IEEE IST, pp. 1–6, 2021