

Exposing the Unseen: Exposure Time Emulation for Offline Benchmarking of Vision Algorithms

Olivier Gamache¹, Jean-Michel Fortin, Matěj Boxan, Maxime Vaidis, François Pomerleau, Philippe Giguère¹

Abstract—Visual Odometry (VO) is one of the fundamental tasks in computer vision for robotics. However, its performance is deeply affected by High Dynamic Range (HDR) scenes, omnipresent outdoor. While new Automatic-Exposure (AE) approaches to mitigate this have appeared, their comparison in a reproducible manner is problematic. This stems from the fact that the behavior of AE depends on the environment, and it affects the image acquisition process. Consequently, AE has traditionally only been benchmarked in an online manner, making the experiments non-reproducible. To solve this, we propose a new methodology based on an emulator that can generate images at any exposure time. It leverages BorealHDR, a unique multi-exposure stereo dataset collected over 10 km, on 55 trajectories with challenging illumination conditions. Moreover, it includes lidar-inertial-based global maps with pose estimation for each image frame as well as Global Navigation Satellite System (GNSS) data, for comparison. We show that using these images acquired at different exposure times, we can emulate realistic images, keeping a Root-Mean-Square Error (RMSE) below 1.78 % compared to ground truth images. To demonstrate the practicality of our approach for offline benchmarking, we compared three state-of-the-art AE algorithms on key elements of Visual Simultaneous Localization And Mapping (VSLAM) pipeline, against four baselines. Consequently, reproducible evaluation of AE is now possible, speeding up the development of future approaches. Our code and dataset are available online at this link: <https://github.com/norlab-ulaval/BorealHDR>

I. INTRODUCTION

Cameras can capture high-resolution details of a scene, at high frame rates, and in a cost-effective manner. Because of this, they are used in many robotic applications, ranging from object detection to localization. One such application is Visual Odometry (VO), which is the task of predicting the displacement of a camera between consecutive images. It is the basis of many vision-based localization algorithms, such as Visual Simultaneous Localization And Mapping (VSLAM), and improving such a task is still an active field of research [1], [2]. In outdoor settings, the quick variations in illumination level and High Dynamic Range (HDR) scenes can severely compromise the performances of VO algorithms [3]. In one second, a car leaving a tunnel can experience an illumination variation over 120 dB [4], while a standard 12-bit channel camera has a theoretical dynamic range of around

^{*}This research was supported by Fonds de Recherche du Québec Nature et technologies (FRQNT) Team grant 254912 and Natural Sciences and Engineering Research Council of Canada (NSERC) DRC Grant through the grant CRD 538321-18, in collaboration with FP Innovations and Resolute Forest Products.

¹Northern Robotics Laboratory, Université Laval, Québec City, Québec, Canada olivier.gamache@norlab.ulaval.ca and philippe.giguere@ift.ulaval.ca

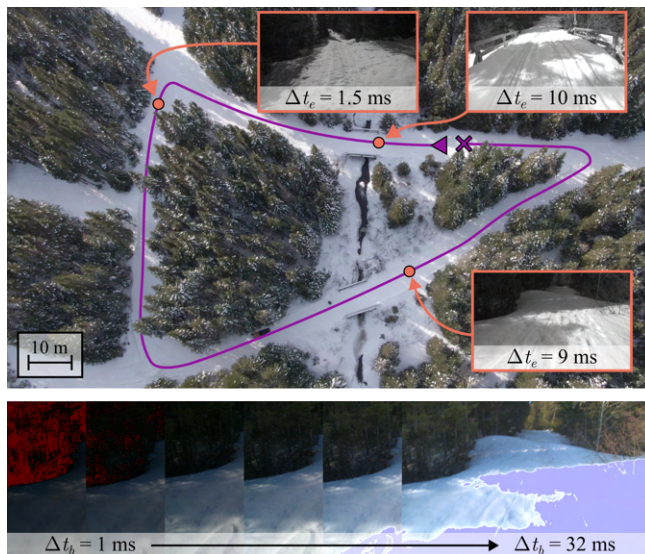


Fig. 1: *Upper image*: Overhead view of a trajectory from BorealHDR dataset, taken in Montmorency Forest, Québec, Canada. The traveled trajectory is shown in purple. Possible emulated exposure times Δt_e along the trajectory are depicted in orange. *Lower image*: Acquired brackets (6), with one used to generate $\Delta t_e = 9$ ms from the upper image. The bracket exposure times Δt_b are $\{1, 2, 4, 8, 16, 32\}$ ms, increasing the dynamic range of our capture by 30 dB or 5 stops. Blue and red colored pixels are over-exposed and under-exposed, respectively.

72 dB. A boreal forest in winter is another example of such HDR environments, where the sun reflected on the snow is highly contrasted with the darkness of the trees. This contrast inevitably leads to saturated pixels on both ends of the spectrum, as highlighted by blue and red colored hues in Figure 1, resulting in the loss of valuable information for VO algorithms. Accordingly, researchers developed Automatic-Exposure (AE) approaches to modify the camera's parameters, such as exposure time and gain, during operation to reduce the impact of the scene's dynamic range on VO [5]–[7].

Unfortunately, given the online nature of exposure control, comparing each method is very challenging [8], [9], as one needs to exactly replicate test conditions for each run, including environment lighting and camera poses. A frequent comparison method for AE algorithms involves capturing images with as many cameras as there are methods to benchmark. As the variety of tested methods grows, managing the experimental setup becomes increasingly complex and expensive. Another alternative involves replicating identical trajectories multiple times for each evaluated AE, which becomes impractical for lengthy trajectories. Moreover, in an uncon-

trolled environment such as the one illustrated in Figure 1, the illumination constantly changes, making it impossible to have a replicable benchmark.

In this work, we propose a novel method for an equitable comparison between AE algorithms applied to robot localization, based on the bracketing technique [10] and realistic image emulation. As illustrated by Δt_e in Figure 1, our emulator framework allows generating images at any desired exposure time for recorded trajectories. This is possible due to our multi-exposure dataset, called BorealHDR, recorded in multi-seasonal conditions in Québec’s forests and on Laval University’s Campus. It is composed of 55 trajectories, totaling more than 5 h of recordings over 10 km. It also includes 3D lidar scans, inertial measurements, and Global Navigation Satellite System (GNSS) data. The novelty of our dataset and our emulation technique allows us to benchmark AE algorithms, feature extraction and VO pipelines **in an offline manner**, as opposed to single-exposure datasets. In short, our contributions are the following:

- 1) A new comparison method for AE algorithms based on an emulation framework that allows to generate realistic images at any plausible exposure time;
- 2) A multimodal dataset that provides stereo images at multiple exposures, alongside 3D map references, in kilometer-scale HDR environments; and
- 3) First offline benchmarking of seven AE algorithms on VO-based experiments.

II. RELATED WORK

We first review the existing methodologies for comparison of AE algorithms. Then, we go over available datasets for VO algorithms and their limitations in the context of testing AE, highlighting the advantages of our dataset. Finally, we discuss important state-of-the-art AE algorithms that will be used for the first benchmarking on a large VO dataset.

A. Methodologies for AE Algorithms Comparison

The nature of AE algorithms makes their comparison very challenging, as they actively change camera parameters during execution. Here, we present state-of-the-art AE benchmarking methods that address this issue, and categorize them between static and moving cameras approaches.

Static Cameras – One way to evaluate AE methods is to fix a camera and acquire multiple pictures at different exposure times. Shim *et al.* [11] used a surveillance camera to collect images with 210 combinations of exposure parameters, six times a day, to compare their new AE method with others. Zhang *et al.* [5] developed an 18 real-world static scenes dataset, where images at varying exposure times were collected in each scene. Kim *et al.* [12] collected a dataset made of 1000 images with multiple different exposures, covering indoor scenes and outdoor scenes. Shin *et al.* [8] were the only ones using a stereo camera to collect a static scene dataset. Multi-exposed static scene images allow comparing some proxies of VO such as single image feature detection.

However, complete trajectories, as proposed in our multi-exposure dataset, are required to benchmark AE algorithms on complete VO pipelines.

Moving Cameras – Three trends can be observed using cameras in motion to compare AE algorithms: simulation-based methods, multi-camera methods, and multi-trajectory methods. An example of *simulation-based methods* was used by Zhang *et al.* [5], which used the Multi-FoV synthetic dataset [13], to simulate multiple versions of an image, but with different exposition levels. Similarly, Gomez-Ojeda *et al.* [14] trained a neural network to produce images with higher gradient information using a synthetic dataset, where they simulated 12 different exposures. Although these techniques reduce dataset development time, they face the *Sim2Real* gap [15], in opposition to our approach, which relies only on real-world images. *Multi-camera methods* are the most widespread for AE algorithms comparison, leveraging a hardware solution for effective data collection. In this setup, two or three cameras are typically installed on a mobile platform [5], [6], [9], [16]. This allows exact comparison since all AE methods are executed simultaneously, thus facing the same conditions. Wang *et al.* [2] used four stereo cameras, facilitating VO comparison, as the stereo depth estimation provides scaling information. An important drawback for these multi-camera methods is that the hardware complexity grows with the number of AE methods tested, rapidly becoming impractical. On the contrary, our approach can be used to compare on an unlimited number of AE methods, at a fixed acquisition cost. *Multi-trajectory methods* consist of repeating multiple times the exact same trajectory using a different exposure control scheme at each iteration. It was used by Bégin *et al.* [7] to develop a gradient AE technique, which was tested using two cameras installed on a motorized 0.16 m² motion table. This allowed for exact repetitions of the same trajectories multiple times, with a ground truth precision near 0.1 mm. To evaluate AE methods in the presence of motion blur, Han *et al.* [1] acquired images from three cameras in motion on a motorized rail. While providing precise ground truth, the rail approach severely restrains the total area covered by the trajectories. It also limits the number of observable environments. A variation of the multi-trajectory method was developed by Kim *et al.* [6]. They drove using stop-and-go maneuvers on a single trajectory, where they stopped six times in total. At each stop, they collected images at multiple exposure times. This technique restricts the number of images that can be acquired, since any changes in the environment would corrupt the benchmark.

In our case, we do not make any static assumption, and we are able to compare AE methods using standard VO pipelines. Our dataset has the advantage of being versatile and independent of the tested solutions, resulting in better replicability, even if new algorithms are developed.

B. Datasets for VO

Several datasets were collected aiming at improving VO against challenging illumination conditions. The KITTI [17] and the Oxford [18] datasets both acquired stereo cameras,

lidars, Inertial Measurement Unit (IMU), and GNSS data, with Oxford offering different weathers, seasons, and illuminations. The North Campus dataset [19] also acquired urban stereo images, changing between indoors and outdoors, on a 15-month range, using a segway robotic platform. The UMA-VI dataset [20] had for main purpose to acquire HDR images with a large number of low-textured scenes. They used a handheld custom rig equipped with cameras and IMU, but they only provided ground truth through loop closure error. Closer to our dataset environment, TartanDrive [21] and FinnForest [22] both acquired off-road data. The TartanDrive dataset contains seven proprioceptive and exteroceptive modalities used for reinforcement learning purposes, including stereo images. Although simpler, the FinnForest dataset is also composed of stereo images in summer and winter, showing the same forest landscapes under multiple conditions. From the papers described in Section II-A, only [7] and [8] published their dataset. While the presented datasets allowed great improvements of VO algorithms, ours is complementary by providing the full dynamic range of scenes through exposure times cycling. Combined with our emulation framework, we unlock the possibility to select the exposure time *during playback*, expanding the realm of camera parameters algorithms evaluation.

C. AE Algorithms

Vision-based localization necessitates sufficient contrast in images to maintain accuracy and robustness. This, in turns, depends heavily on the camera’s AE algorithm and its accompanying target metric. For instance, Shim *et al.* [11] used image gradient magnitude as such metric. Their exposure control scheme generates seven synthetic versions of the latest acquired image, simulating different exposure levels, to identify the next exposure value maximizing their metric. The AE algorithm proposed by Zhang *et al.* [5] instead sorts the gradient level of each pixel, and applies a weight factor based on their percentile value. By combining their quality metric and the Camera Response Function (CRF), they predict the best next exposure value. Kim *et al.* [12] developed an image quality metric based on the gradient level and Shannon entropy [23], used to detect saturation. To demonstrate the benchmarking capabilities of our approach, we provide an implementation of the above methods, which are often used for comparison [2], [7], since they cover main aspects of image quality for localization algorithms. Other techniques exist, such as Shin *et al.* [8], who takes into account the Signal-to-Noise Ratio (SNR), but we did not implement it, since it is mostly based on camera gain.

III. METHODOLOGY

In this section, we detail the selected approaches for the development of our system, namely the emulation technique, the AE implementations, and the lidar-based reference trajectories. Details on how our system can be used for benchmarking are presented in Section IV.

A. Emulation Technique

From the real images acquired using the bracketing technique, we are able to emulate an image at any other realistic exposure time. Our emulation method is based on the image acquisition process, which maps the scene radiance E to the image pixel values $I(x)$, using the vignetting $V(x)$, the exposure time Δt , and the CRF $f(\cdot)$. This process can be expressed using the following equation [24]:

$$I(x) = f(\Delta t V(x) E). \quad (1)$$

From Equation 1, the relationship between two images I_{source} and I_{target} and their respective exposure times Δt_{source} and Δt_{target} can be defined as

$$I_{\text{target}} = f\left(\frac{\Delta t_{\text{target}}}{\Delta t_{\text{source}}} \cdot f^{-1}(I_{\text{source}})\right), \quad (2)$$

where $f^{-1}(\cdot)$ is the inverse CRF [25]. To estimate $f(\cdot)$ and $f^{-1}(\cdot)$, we take multiple images of a static scene at several exposure times, allowing to capture the whole dynamic range at fixed radiance, following [26]. With $f^{-1}(\cdot)$ and Equation 2, it is thus possible to emulate any targeted exposure times I_{target} by using a known image with an exposure time Δt_{source} . Note that, for the sake of simplicity, an image taken from one exposure time in the bracketing cycle will be called *bracket*.

Considering that the data is acquired while moving, the brackets are not taken at the same time nor position. Therefore, we decided not to interpolate between the brackets, since it would create artifacts in the resulting image. Instead, we select the most appropriate I_{source} from the available brackets to emulate I_{target} . The selection process takes into account the distance between Δt_{source} and Δt_{target} , and the amount of saturated pixels in I_{source} , since these do not contain any usable information. Also, if we select a bracket such as $\Delta t_{\text{target}}/\Delta t_{\text{source}} > 1$, the result will be an image with higher pixel values, which means that the noise in the signal will also increase. Therefore, it is preferable to select I_{source} that has a higher exposure time compared to our desired I_{target} to maximize the SNR.

Based on these considerations, we developed a simple selection method named HIGHERNOSAT. The first step is to find the two closest brackets Δt_{bl} and Δt_{bh} , that bound Δt_{target} , such that:

$$\Delta t_{\text{bl}} < \Delta t_{\text{target}} < \Delta t_{\text{bh}}. \quad (3)$$

Then, for each of the two brackets, we calculate the amount of image saturation, which corresponds to the number of pixels with values of 0 and 4095, for our 12-bit channel images. If the saturation level of the higher bracket is below 1%, we select it as the best candidate, otherwise, we pick the lower one. Finally, if Δt_{target} is outside the range of available Δt_{source} , we select the closest bracket.

B. Implementation Details of Compared AE Algorithms

Based on Section II-C, we implement and benchmark three state-of-the-art AE methods: M_{Shim} [11], M_{Zhang} [5],

and M_{Kim} [12]. These methods are not open-source, thus they were all implemented based on our understanding of the papers. M_{Zhang} suffers from instabilities coming from the CRF estimation, which were partly resolved using the description in [8]. The exposure control scheme of M_{Kim} used a Gaussian Process (GP), which sparsely sweep the camera exposure parameters until convergence. In our implementation, we use a sliding window on the trajectory to only consider the most recent images in the GP training. This can cause steep changes in the desired exposure time, since the exploration algorithm of the GP needs to cover a wide range of values to converge. We also implemented four baseline methods, corresponding to the typical AE algorithms. The first baseline algorithm is a fixed exposure time approach M_{fixed} . This emulated exposure time is selected once, at the beginning of each sequence, by using a brightness target of 50%. The three other baselines are variable exposure algorithms, seeking to keep the average brightness target to 30%, 50%, and 70% of the 12-bit depth range. They are named respectively $M_{30\%}$, $M_{50\%}$, $M_{70\%}$. These percentages were chosen as to cover a wide range of exposure.

C. Reference trajectories

In addition to the cameras, the platform is equipped with a lidar and a GNSS antenna, which both provide useful information to estimate the traveled trajectories. In boreal forests, Baril *et al.* [27] demonstrated that GNSS signals are unreliable, due to attenuation caused by the dense vegetation canopy. Therefore, our reference trajectories were generated by employing a low-drifting lidar-inertial-Simultaneous Localization And Mapping (SLAM) variant [28], based on the libpointmatcher library¹. This method registers deskewed point clouds, at a rate of 10 Hz, into a dense 3D map of the environment. It relies on the on-board lidar and is illumination-independent, with median error values around 1% of the total trajectory length. Note that, we do not assume that the reference trajectories are equivalent to the ground truths, but that they provide an adequate precision for comparison with our VO pipeline, as explained in Section IV-D.2.

IV. RESULTS

This section first describes our acquisition platform, before evaluating the performances of the emulation method. Then, we elaborate our data collection pipeline, and finally, conduct the first online AE benchmarking on large VO-based trajectories.

A. Experimental Setup

The BorealHDR dataset was collected using a homemade water-resistant acquisition backpack, depicted in Figure 2. The enclosure is a Pelican case 1510 containing a battery and a Jetson AGX Xavier Development kit embedded computer. The camera sensors are two Basler a2A1920-51gcPRO, in a stereo-calibrated configuration with a baseline of 18 cm, and hardware-triggered by an STM32 microcontroller ensuring



Fig. 2: Picture of the developed backpack for the dataset acquisition. Main components are identified as follows: ① Two Basler a2A1920-51gcPRO cameras, ② Xsens MTI-30 IMU, ③ VLP16 3D lidar, ④ Emlid Reach RS+ GNSS receiver, and ⑤ Ubiquiti UniFi UAP-AC-M wifi antenna.

precise exposure times and sensor synchronization. In addition, the platform is equipped with a Velodyne VLP-16 lidar, an Xsens MTI-30 IMU, and an Emlid Reach RS+ GNSS receiver. Similar to [29], the extrinsic calibrations between the sensors are given by the 3D model of the platform. The intrinsic calibrations for the cameras were also calculated, by using a standard checkerboard. The calibration sequences, and the 3D model are provided with the dataset. The images are acquired at a rate r_{real} of 22 frames per second (FPS) in 12-bit color, using loss-less compression. The image acquisition process cycles through six exposure time values, i.e., $\Delta T_{\text{bracket}} = \{1, 2, 4, 8, 16, 32\}$ ms, yielding an effective emulation rate of $r_{\text{emul}} = 3.66$ FPS. This number of brackets is a compromise between the offline r_{emul} and the emulation error, detailed in Section IV-B. The $\Delta T_{\text{bracket}}$ values were chosen to cover the whole dynamic range of our target environments, while mitigating motion blur. The exposure time is doubled at every step of the cycle, which is equivalent to a one stop increase in photography. The small footprint of the acquisition platform allows collecting data in narrow and hard-to-access spaces for robotic vehicles, which was fundamental for our 10 km dataset in HDR environments.

B. Emulation

To evaluate the performances of our image emulator, we collect five test sequences of 1000 ground truth images with exposure times ranging from 20 μs to 50 ms, in static scenes both indoor and outdoor. Then, we emulate the same images following the method described in Section III-A, and calculate the Root-Mean-Square Error (RMSE) between I_{emul} and I_{GT} for each ground truth exposure. To account for the camera's intrinsic noise, we average the error between 25 consecutive images at the same exposure time, over the whole

¹<https://github.com/norlab-ulaval/libpointmatcher>

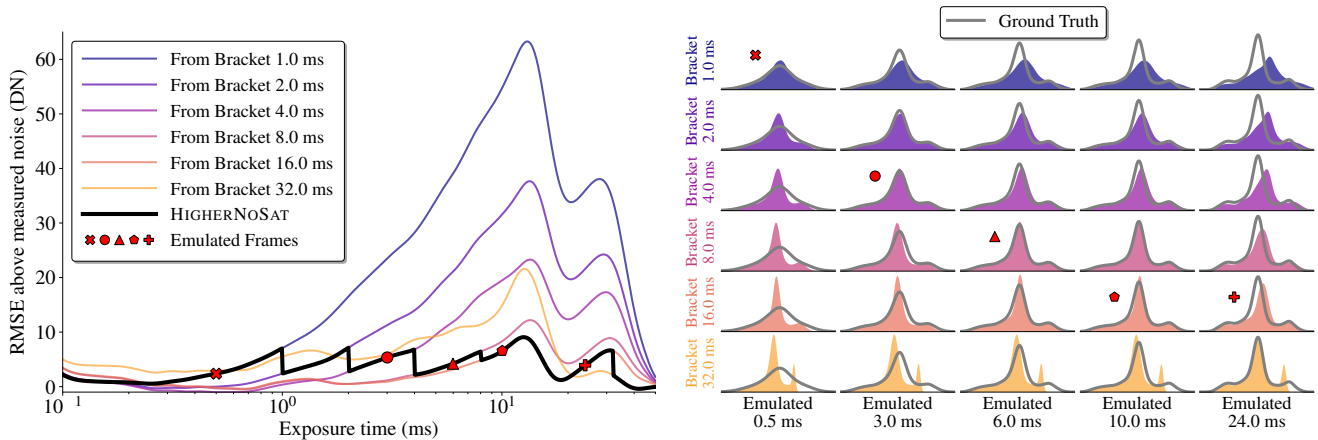


Fig. 3: Validation of our emulation framework using 1000 ground truth images captured with constant illumination, but varying exposure time. **Left:** RMSE curves showing the emulation error if a single bracket was always selected, in colors, compared to our bracket selection method HIGHERNOSAT, in black. Red symbols correspond to the emulated exposure times displayed on the right. **Right:** Qualitative comparison of the distributions of five emulated exposure times (columns) from six different bracket images (rows). The markers are placed next to the selected bracket by HIGHERNOSAT. Ground truth distributions are overlaid in gray for each column.

spectrum, and subtract this noise from our results. The five test sequences serve to validate our bracket selection method HIGHERNOSAT, which does not have access to ground truth images in real world settings. Overall, our tests show that the emulation method maintains a median RMSE of 0.21 % and never surpasses 1.78 %, emphasizing its consistency and accuracy. Note that the sequences are provided with the dataset.

One of the test sequences is detailed in Figure 3, showing emulation performances in controlled lab settings. On the left side, the RMSE curve from HIGHERNOSAT, in black, is compared to the error obtained by selecting a single bracket as I_{source} for the whole range, in colors. It shows that our bracket selection maintains the error close to the lowest value of the six colored curves. The right side plots present a qualitative evaluation of the emulated images from the acquired brackets. Each column exposes the distributions of pixel intensity for one target exposure time, while each row emulates this image from a different bracket. Red markers represent which exposure time in $\Delta T_{\text{bracket}}$ was chosen by our emulator via HIGHERNOSAT, for this image. We observe that our bracket selector consistently picks the closest or second-closest emulated image to the ground truth distribution, overlaid in gray.

C. Dataset Gathering

Our BorealHDR dataset comprises 55 sequences totaling around 10 km for 5 h, and around 2.5 TB of data. Half of the trajectories are loops, which is a common practice to estimate drift in VSLAM datasets. The relatively low r_{emul} implies that the data should be collected at a slow walking speed to minimize displacements between each acquisition cycle. Hence, our average pace is around 2 km/h, which increases the data collection time but does not impact the users of our offline emulation approach. In total, 393 238 images were gathered on four separate days in summer and winter 2023. The winter data, from April 20th to April 21st, was acquired

in the Montmorency boreal Forest near Québec City, Canada. The summer trajectories were collected on the campus of Laval University and on Mont-Bélair, in Québec City, respectively on the 25th and the 27th of September. There are 44 trajectories from Montmorency boreal Forest, four acquire on the university campus, and seven from Mont-Bélair, each site respectively totaled 7.5 km, 1 km, and 1.5 km. We chiefly focused the acquisitions for HDR scenes, either by capturing trees and snow in the camera frame for April’s data, or shadows and sunlight’s area, in the September’s ones. In addition to HDR scenarios, multiple trajectories contain challenging conditions for VO, such as sun glares and texture-less environment from immaculate snow and a lake steady water. In winter, we used snowshoes to get to remote locations, to collect not only different scene illuminations, but also various 3D structures, allowing for a richer collection of test scenarios.

D. Benchmark

The main contribution of our paper is to unlock *offline* testing of active vision methods that could only previously be tested in an *online* manner. To this effect, we conduct two series of tests to benchmark the methods explained in Section III-B, namely (1) *feature tracking and uniformity*, and (2) *stereo VO*, both highlighting the impact of each AE algorithm in key aspects of a VSLAM pipeline.

1) *Feature Tracking and Uniformity:* For feature-based VO algorithms, the performance is dependent on the quality and quantity of detected keypoints. Therefore, we evaluate feature detection for each implemented AE algorithm on our recorded dataset, over three distinct tests. We selected the Scale-Invariant Feature Transform (SIFT) [30] feature-detector, since it is pervasive in the literature, and the most accurate according to [31]. Learning-based feature-detectors such as SuperPoint [32] could be considered for evaluation. However, their performance is sensitive to the domain gap

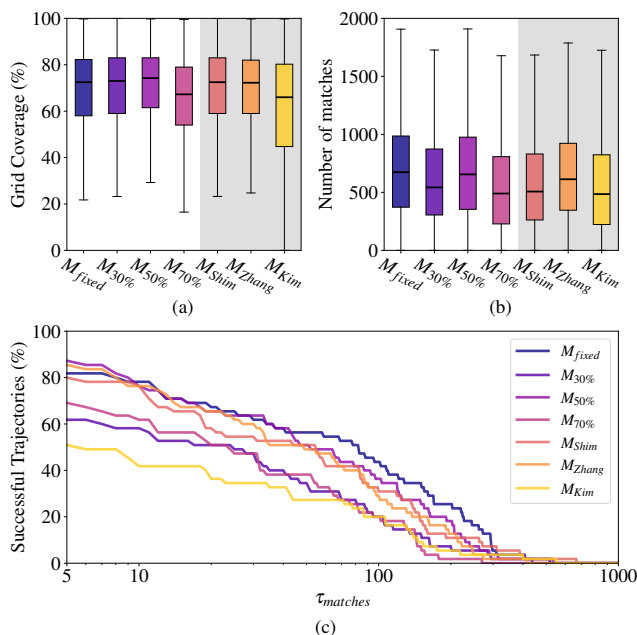


Fig. 4: SIFT features analysis. (a) Uniformity of detected keypoints for an image divided into a 20×20 grid. (b) Number of match features between a pair of images for all AE metrics. The gray shading in (a) and (b) is to highlight the state-of-the-art AE methods. (c) Percentage of successful trajectories based on the number of matches detected. If one image contains less matches than τ_{matches} , the sequence is marked as not successful.

between training and test data, which could bias results. Consequently, SIFT provides for an equitable analysis of the methods. The first test consists of estimating the uniformity of the detected keypoints, by dividing each image into a 20×20 grid and assessing its occupancy rate. A uniform distribution of features is presumed to be a good proxy for VO algorithm performance. The results are displayed in Figure 4a, which shows that most implemented AE methods yield a similar grid coverage. We observe that $M_{70\%}$ and M_{Kim} provide slightly reduced uniformity, with median values respectively 67.3 % and 66.0 % under the average of the other combined methods. In Figure 4b we show that we can also obtain the number of detected features that are matched between consecutive images, again for all implemented AE methods. M_{fixed} is the one tracking the highest number of features between two images, with a median of 674 matches. This result was expected since the exposure time is calibrated when starting each sequence, and the lack of exposure change keeps a better flow between consecutive images. Now, looking at full trajectories, we define a success criterion corresponding to a minimal number of matches τ_{matches} between each consecutive image pairs. If, for a given AE algorithm and trajectory, this criterion is always true, then the sequence is considered successful. Accordingly, we evaluate the percentage of successful trajectories with $\min(\tau_{\text{matches}}) = 5$, the minimum number of matches for motion estimation [33], and the results are shown in Figure 4c. We observe that the overall robustness of the AE algorithms decreases rapidly. For $\tau_{\text{matches}} = 100$, which is largely under any first quartile

from Figure 4b, $M_{30\%}$, $M_{70\%}$, and M_{Kim} shows the poorest performance, completing 20 % of the dataset sequences successfully, while the best method, M_{fixed} , achieves 44 % of success. This clearly shows the challenge that represents our dataset for VO.

Many of the developed benchmarking techniques for AE algorithms would have been able to conduct our first two experiments. However, the third one could only have been done using methodologies based on moving camera, since we evaluate the features on trajectories. More importantly, our approach enables entirely reproducible testing, offline. Consequently, we can always add more AE methods to the benchmarks, without having to collect new data.

2) *Stereo Visual Odometry (VO)*: We also investigate the impact of AE methods on VO, which is a key component of VSLAM. We implemented a minimal stereo VO pipeline based on *OpenCV* [34] using SIFT [30] for feature detection. The *evo*² library is used to calculate the Relative Pose Error (RPE) [35] for each AE algorithm. Our reference trajectories are generated using the lidar-inertial-SLAM system, as explained in Section III-C. For an equitable comparison, the scale alignment feature from the *evo* library was applied on the VO trajectories. An example using one trajectory from our dataset is illustrated in Figure 5, along with the 3D map. The VO algorithm was unsuccessful for 11 of the 55 trajectories, resulting in 20 % of failure cases, highlighting the challenge that represents our new dataset BorealHDR for VO. Multiple of the failed trajectories contain sun glares and illumination variation, which could explain the poor performances.

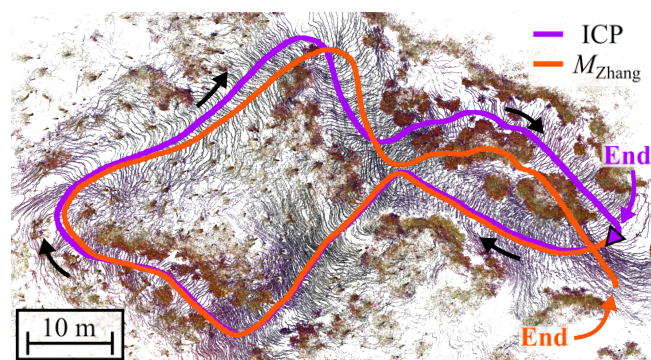


Fig. 5: Qualitative example illustrating the trajectory `backpack_2023-04-21-10-57-59` from our BorealHDR dataset. The lidar-based trajectory (purple) and M_{Zhang} 's emulated VO result (orange) are illustrated superimposed on the generated lidar 3D map. The higher drift of the VO localization can be highlighted with the loop closure done in this forest environment. Black color represents the ground, and other colors highlight structures in the environment.

The RPE is calculated on all the trajectories for which the stereo VO pipeline converged, and is displayed in Figure 6. We observe that all the AE methods have a minimal translation error above 4.21 %, obtained by $M_{50\%}$. The least performing method was M_{Kim} , which has a minimal error

²<https://github.com/MichaelGrupp/evo>

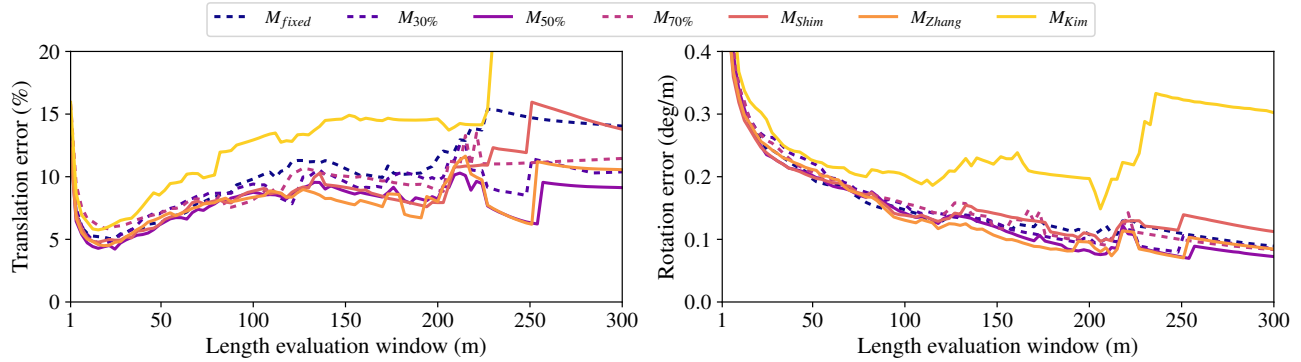


Fig. 6: RPE over all the trajectories as a function of the trajectory length, for the seven implemented AE algorithms. *Left*: Median translation error. *Right*: Median rotation error. The line styles depict three clusters, based on the translation performances, going from the top with solid, dashed, and then solid again.

of 5.75%. This is caused by the GP’s exploration step, as explained in [Section III-B](#). The rotational error has similar results, where M_{Zhang} and $M_{50\%}$ are the two best methods of the benchmark, with a minimal error of 0.07 deg/m. Even though M_{fixed} was among the best methods for feature tracking, its unresponsiveness to illumination changes results in an error of 5.1% in translation and 0.09 deg/m in rotation, when applied to VO. From this experiment, all the tested AE algorithms performed similarly on the dataset, with a small advantage for gradient-based methods. In summary, our new developed approach allows for the first comparison of AE methods directly on a large VO dataset, in an offline manner.

V. CONCLUSION AND FUTURE WORK

In this work, we proposed an emulator framework based on a multi-exposure dataset, which allows comparing AE algorithms in an offline and reproducible manner. Our dataset, BorealHDR, contains 55 trajectories combining for 10 km, focusing on collecting images in HDR scenes taken in a snowy boreal forest environment. In addition to the 12-bit color stereo images, we also provide lidar-inertial-SLAM based pose for each image and 3D map for each sequence, based on lidar data. We have shown the versatility of our methodology by benchmarking seven AE algorithms, of which three were from recent works. We concluded that our emulation approach is an efficient solution for offline benchmarking of active algorithms, such as AE, by making experiments on multiple key elements affecting VSLAM pipelines. In the future, presuming improvements in processing time, our system could be used in real-time to emulate images based on VSLAM performances. Similarly, we also plan to design novel AE approaches that take into account the pose uncertainty estimates obtained by the VSLAM pipeline, which were previously difficult to develop without a flexible testing methodology.

REFERENCES

- [1] B. Han, Y. Lin, Y. Dong, H. Wang, T. Zhang, and C. Liang, “Camera Attributes Control for Visual Odometry With Motion Blur Awareness,” *IEEE/ASME Transactions on Mechatronics (TMECH)*, vol. 28, no. 4, pp. 2225–2235, 2023.
- [2] Y. Wang, H. Chen, S. Zhang, and W. Lu, “Automated camera-exposure control for robust localization in varying illumination environments,” *Autonomous Robots*, vol. 46, no. 4, pp. 515–534, 2022.
- [3] G. Chahine and C. Pradalier, “Survey of Monocular SLAM Algorithms in Natural Environments,” in *Conference on Computer and Robot Vision (CRV)*, 2018.
- [4] J. Tomasi, B. Wagstaff, S. L. Waslander, and J. Kelly, “Learned Camera Gain and Exposure Control for Improved Visual Feature Detection and Matching,” *IEEE Robotics and Automation Letters (RA-L)*, vol. 6, no. 2, pp. 2028–2035, 2021.
- [5] Z. Zhang, C. Forster, and D. Scaramuzza, “Active exposure control for robust visual odometry in HDR environments,” in *IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 3894–3901.
- [6] J. Kim, Y. Cho, and A. Kim, “Proactive Camera Attribute Control Using Bayesian Optimization for Illumination-Resilient Visual Navigation,” *IEEE Transactions on Robotics (T-RO)*, vol. 36, pp. 1256–1271, 2020.
- [7] M.-A. Bégin and I. Hunter, “Auto-Exposure Algorithm for Enhanced Mobile Robot Localization in Challenging Light Conditions,” *Sensors*, vol. 22, p. 835, 3 2022.
- [8] U. Shin, J. Park, G. Shim, F. Rameau, and I. S. Kweon, “Camera Exposure Control for Robust Robot Vision with Noise-Aware Image Quality Assessment,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 1165–1172.
- [9] I. Shim, T. H. Oh, J. Y. Lee, J. Choi, D. G. Choi, and I. S. Kweon, “Gradient-based camera exposure control for outdoor mobile platforms,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, pp. 1569–1583, 6 Jun. 2019.

- [10] M. Gupta, D. Iso, and S. K. Nayar, "Fibonacci Exposure Bracketing for High Dynamic Range Imaging," in *IEEE International Conference on Computer Vision (ICCV)*, 2013.
- [11] I. Shim, J.-Y. Lee, and I. S. Kweon, "Auto-adjusting camera exposure for outdoor robotics using gradient information," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2014, pp. 1011–1017.
- [12] J. Kim, Y. Cho, and A. Kim, "Exposure Control Using Bayesian Optimization Based on Entropy Weighted Image Gradient," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 857–864.
- [13] Z. Zhang, H. Rebecq, C. Forster, and D. Scaramuzza, "Benefit of large field-of-view cameras for visual odometry," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 801–808.
- [14] R. Gomez-Ojeda, Z. Zhang, J. Gonzalez-Jimenez, and D. Scaramuzza, "Learning-based image enhancement for visual odometry in challenging HDR environments," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 805–811.
- [15] S. Höfer, K. Bekris, A. Handa, *et al.*, "Sim2Real in robotics and automation: Applications and challenges," *IEEE transactions on automation science and engineering*, vol. 18, no. 2, pp. 398–400, 2021.
- [16] I. Mehta, M. Tang, and T. D. Barfoot, "Gradient-Based Auto-Exposure Control Applied to a Self-Driving Car," in *Conference on Computer and Robot Vision (CRV)*, 2020, pp. 166–173.
- [17] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2012, pp. 3354–3361.
- [18] D. Barnes, M. Gadd, P. Murcutt, P. Newman, and I. Posner, "The Oxford Radar RobotCar Dataset: A Radar Extension to the Oxford RobotCar Dataset," in *IEEE International Conference on Robotics and Automation (ICRA)*, May 2020, pp. 6433–6438.
- [19] N. Carlevaris-Bianco and R. M. Eustice, "Learning visual feature descriptors for dynamic lighting conditions," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2014, pp. 2769–2776.
- [20] D. Zuñiga-Noël, A. Jaenal, R. Gomez-Ojeda, and J. Gonzalez-Jimenez, "The UMA-VI dataset: Visual-inertial odometry in low-textured and dynamic illumination environments," *The International Journal of Robotics Research*, vol. 39, pp. 1052–1060, 9 Aug. 2020.
- [21] S. Triest, M. Sivaprakasam, S. J. Wang, W. Wang, A. M. Johnson, and S. Scherer, "TartanDrive: A Large-Scale Dataset for Learning Off-Road Dynamics Models," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2022, pp. 2546–2552.
- [22] I. Ali, A. Durmush, O. Suominen, *et al.*, "Finn-Forest dataset: A forest landscape for visual SLAM," *Robotics and Autonomous Systems*, vol. 132, p. 103610, Oct. 2020.
- [23] C. E. Shannon, "A mathematical theory of communication," *The Bell system technical journal*, vol. 27, no. 3, pp. 379–423, 1948.
- [24] P. Bergmann, R. Wang, and D. Cremers, "Online photometric calibration of auto exposure video for real-time visual odometry and slam," *IEEE Robotics and Automation Letters*, vol. 3, pp. 627–634, 2 Apr. 2018.
- [25] M. Grossberg and S. Nayar, "Determining the camera response from images: What is knowable?" *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, pp. 1455–1467, 11 Nov. 2003.
- [26] J. Engel, V. Usenko, and D. Cremers, "A photometrically calibrated benchmark for monocular visual odometry," in *arXiv:1607.02555*, 2016.
- [27] D. Baril, S.-P. Deschênes, O. Gamache, *et al.*, "Kilometer-scale autonomous navigation in subarctic forests: Challenges and lessons learned," *Field Robotics*, vol. 2, no. 1, pp. 1628–1660, Mar. 2022.
- [28] V. Kubelka, M. Vaidis, and F. Pomerleau, "Gravity-constrained point cloud registration," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 4873–4879.
- [29] Lv, Jiajun and Xu, Jinhong and Hu, Kewei and Liu, Yong and Zuo, Xingxing, "Targetless Calibration of LiDAR-IMU System Based on Continuous-time Batch Estimation," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 9968–9975.
- [30] D. G. Lowe, "Object recognition from local scale-invariant features," in *IEEE International Conference on Computer Vision*, 1999, pp. 1150–1157.
- [31] S. A. K. Tareen and Z. Saleem, "A comparative analysis of sift, surf, kaze, akaze, orb, and brisk," in *IEEE International conference on computing, mathematics and engineering technologies (iCoMET)*, 2018, pp. 1–10.
- [32] D. DeTone, T. Malisiewicz, and A. Rabinovich, "Superpoint: Self-supervised interest point detection and description," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2018, pp. 224–236.
- [33] D. Nistér, "An efficient solution to the five-point relative pose problem," *IEEE transactions on pattern analysis and machine intelligence*, vol. 26, no. 6, pp. 756–770, 2004.
- [34] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.
- [35] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of rgb-d slam systems," in *IEEE/RSJ international conference on intelligent robots and systems*, 2012, pp. 573–580.