Using buildings proximity information in UAV global localization over satellite images

Gean Stein[1] Diego Pittol[1] Mathias Mantelli[1] Mariana Kolberg[1] Edson Prestes[1] Renan Maffei[1]

***Abstract*— Unmanned Aerial Vehicles (UAVs) depend on localization systems to safely and efficiently perform their tasks, usually relying on GPS. However, this setup is not problem-free, and having only one source of measurement may be catastrophic to the UAVs' operation. To avoid this kind of issue, this paper proposes a visual UAV localization system as an extra source of pose estimation in case of GPS malfunctioning. The novelty of our system is its measurement model used in the Monte Carlo Localization, which relies on a new descriptor called NBD-BRIEF, based on the Nearest Building Distance (NBD) information obtained at the vehicle position. The building footprint is extracted from the UAV images using a convolutional network. Such semantic information is more stable and less sensitive to light and color changes, making the method generally more robust than approaches purely based on color matching. The experiments have shown that our NBD-BRIEF descriptor performs better than competing approaches to the same problem. And the results indicate that our visual UAV localization system based on NBD-BRIEF can properly estimate the UAV's pose in three different flights, while other localization methods we compared failed.**

I. Introduction

In recent years, unmanned aerial vehicles (UAVs) have dealt with many different tasks in robotics and automation. Medical product transport [1], agriculture [2], and even Mars exploration [3] are just a few examples of UAV deployments. Across all applications, it is vital to precisely estimate the UAV's pose, mainly for the cases in which it is autonomously operating. The majority of UAVs rely on their embedded GPS sensor to estimate their position with the aid of the global navigation satellite systems (GNSS). Even though GNSS is widely used, issues in its signal, such as instability and multi-path propagation [4], [5], may increase the uncertainty of the position estimation. Besides, UAVs can be hijacked by spoofed GNSS signal [6], [7], which may be catastrophic to the safety of the task.

Therefore, an alternative system is required as a redundant source of pose estimation. Vision-based localization systems are one of the most popular solutions to play the role of a secondary pose estimation system when the GPS fails. Not only are cameras more portable and have a lower cost when compared with other solutions, such as laser rangefinders, but visual localization techniques can also be used in various situations.

*This work was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) [Finance Code 001] and in part by the Brazilian National Council for Scientific and Technological Development (CNPq).

[1]Institute of Informatics, Universidade Federal do Rio Grande do Sul, Porto Alegre, Brazil stein.gean@gmail.com, dpittol, mfmantelli, mariana.kolberg, prestes, rqmaffei@inf.ufrgs.br

(a) (b) (c)

Fig. 1: Information about buildings in urban areas (**b**) tends to be a more reliable source of information than pure visual information, (**a**), which suffers variations due to changes in weather, season, etc. However, such information ends up generating large homogeneous regions, unduly increasing the similarity between different regions. On the other hand, building proximity i.e. the distance to the nearest building, (**c**), is a measure that varies along the environment and can be used to improve the localization process. In (b), buildings are represented with white pixels. In (c), the whiter a pixel is, the further it is from the buildings' edges.

Visual localization for UAVs is a challenging problem for several reasons. The main ones are related to comparing the UAV sensor readings, usually 2D images, and the map of the environment, frequently a 2D satellite image. The satellite image may be outdated concerning the flight date, making the UAV images and the satellite map considerably different. Besides, as the UAV flies outdoors during the pose estimation, there may be illumination changes across the sequence of UAV images or even perspective effects in the buildings and high structures.

Couturier and Akhloufi [5] present an excellent overview of several works on visual UAV localization in general and discuss how they deal with the challenges of this problem. Cited works [8], [9], [10] range from visual odometry that tracks the UAV by the sequence of its images to feature-based approaches that compare the UAV images against multiple satellite image patches.

Among the methods based on matching satellite image maps and UAV images, one that showed excellent robustness was abBRIEF, proposed by Mantelli et al. [11]. The main contribution of their work is a measurement model for the Monte Carlo Localization (MCL) system using the so-called abBRIEF descriptor, which extends the Binary Robust Independent Elementary Features (BRIEF) descriptor [12]. Their experiments show that their system correctly estimates the UAV's pose in multiple flights and maps. However, the weakness of their system is homogeneous areas, such as forests and pastures. Since their abBRIEF descriptor relies on the color information to compute the image signature, their

system cannot efficiently estimate the UAV's pose when it flies over a region represented mainly by a single color.

An alternative way of approaching the localization problem being less influenced by issues arising from color variations in the images, is the use of semantic information that varies little over time, such as buildings, roads, among others. Choi and Myung [13] proposed a visual UAV localization system using buildings information obtained through image segmentation, along with a concept called Building Ratio. They compare the UAV image against patches of the map, searching for possible matches. The system continues until there is a convergence and the UAV position is estimated. A weakness of the Building Ratio-based approach is that it compresses the image information too much, which hinders convergence in more complex environments. Additionally, the experiments only considered flights at fixed heights.

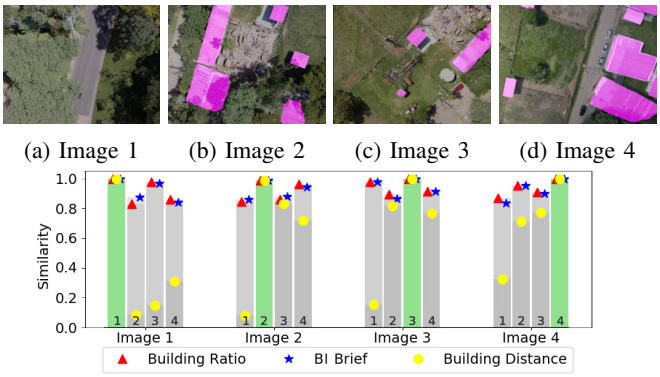
Despite the valuable work proposed by the research community, there is still room for improvement in the field of visual UAV localization. In this paper, we propose a new vision-based UAV localization system that is based on a less varied source of information than the pure colors of an image. The presence of buildings in a given region of an urban area tends to pass through far fewer changes than purely visual information such as light, color, or vegetation of the same area during the day, throughout the seasons, or over the years. However, as illustrated in Fig. 1b, the information of what is a building and what is not usually is not quite descriptive due to its homogeneous nature; thus, our method proposes using a measure of proximity to buildings, which varies gradually between neighboring pixels, as shown in Fig. 1c. Our main contributions are:

- NBD-BRIEF - an image descriptor based on the distances to the nearest buildings in the image;
- a visual UAV localization framework based on NBD-BRIEF, that uses as input a reference map containing building footprints, and RGB images taken by an UAV and processed with a convolutional neural network to classify building information.

This paper is organized as follows. Section II explores the importance of a good measurement model for a UAV localization system, giving a context of the importance of MCL for visual UAV localization and the impact of different ways of matching images. Section III details the proposed approach. Section IV describes the experiments, and Section V, the conclusion.

II. The Importance of a Good Measurement Model in the UAV Localization Problem

The general idea of MCL, widely used in localization problems [14], is to spread particles (i.e., virtual copies of the robot) on the environment map to perform the pose estimation through a cycle of prediction and correction. A motion model is used in the prediction step by moving the particles according to the robot's odometry information. Then, the state is corrected using a measurement model, which compares the robot's sensor readings with the particles' readings. The more similar the particle's readings are to the robot's,



(e) Similarities between images using 3 different approaches

Fig. 2: Analysis of three different similarity measurements (Building Ratio, Binary-Input Brief and Building Distance) computed between pairs of four images containing building information (Images 1 to 4). The original images captured by the UAV are shown in the background (darker colors). Still, only the building information (pink) or its absence is used by the methods to compute the similarities.

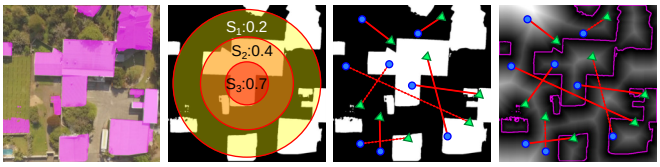


Fig. 3: Different descriptors considering the building information shown in (a): (b) **Building ratio**: density of building cells inside a kernel (results with three different kernel sizes are shown); (c) **Binary-Input BRIEF** (Building vs Non-building info): comparisons of binary values extracted from random pairs of points; and (d) **BRIEF w/ Building Distance** info: likewise, but comparing values of distance to borders of buildings.

the higher the particle's weight required for sampling a new set of particles. In the context of UAV localization, MCL is one of the most popular algorithms [5], with a satellite image as the reference map. Thus, the measurement model must compare the image taken by the UAV with patches of the reference map simulating the view of particles.

RGB images captured by a UAV contain a considerable amount of information, representing the environment that the UAV flies over. On the other hand, segmented images that classify the environment according to only one class, such as buildings, have two possible pixel values: true or false. Thus, there is a trade-off between relying on information susceptible to variations and not having enough information to perform the image matching efficiently.

Fig. 2 shows the comparisons between different matching strategies applied to binary images segmented as building/non-building. Four images were chosen for testing containing different configurations of buildings, highlighted in pink. Three different approaches were tested matching all pairs of images, with results shown in Fig. 2e. The selected techniques are illustrated in Fig. 3, and described next.

The first strategy analyzed is the application of the Build-

ing Ratio concept [13], illustrated in Fig. 3b. It computes a kernel density estimate indicating the ratio of an area associated with buildings over the total area within a kernel centered in the middle of the image[1]. It is possible to observe in Fig. 2 that very different configurations of buildings have similar building ratios because the space without buildings in the images is much larger than the opposite. Therefore, it is a strategy that serves to differentiate well only substantial changes between regions.

The second strategy analyzed that we called Binary-Input BRIEF and is illustrated in Fig. 3c, is a variation of the abBRIEF matching algorithm [11], but using the binary image of building classification as a source. The descriptor is constructed by comparing pairs of random pixels within the same image and establishing whether the first element of the pair has a greater intensity than the second. To compare two images, the descriptor is generated in each image, with the same pairs of points, and the difference between the descriptors is the similarity result. This type of strategy works well for colored images [11], as any variations in colors tend to be consistent within the image itself (e.g., everything is brighter or with greater contrast). On the other hand, in binary images, the regions are mostly homogeneous, so there is a high chance that distinct pairs of points will generate the same result. We can see that since most of every image does not contain buildings, most of the pairs of points will tend to have the same value, and the generated descriptors become very similar.

Finally, the third strategy analyzed, illustrated in Fig. 3d, is what we are proposing in this work: the use of proximity to buildings instead of simply considering whether or not the region is a building. The idea is that by using the proximity to buildings data, i.e., the distance to the nearest building, we have more information to use in the matching process while ensuring that such information is as reliable as the original binary image. With this strategy, the differences in the image-matching result are much more pronounced, as shown in Fig. 2e. For example, the image in Fig. 2a, which does not contain buildings, becomes very different from the others. Furthermore, even the three remaining images have a more significant difference among them with this strategy than evaluating them with the previous ones, which is excellent for improving the quality of the localization estimate.

Compared to the previous descriptors, especially to the first one, the distance-based descriptor is much more sensitive, with the disadvantage of being less robust to noise. However, this is a small drawback compared to the disadvantage of using so little information, like the other methods, that could prevent successful localization.

III. Proposal - Localization using Buildings Proximity Information

A. Framework Overview

In this paper, we propose a global localization approach using MCL, that compares images obtained by a UAV looking downward with a satellite image of the region where the robot is located. The weighting step of the MCL[2] is based on constructing a BRIEF-based descriptor obtained from images describing the proximity to buildings. Fig. 4 presents an overview of the proposed framework.

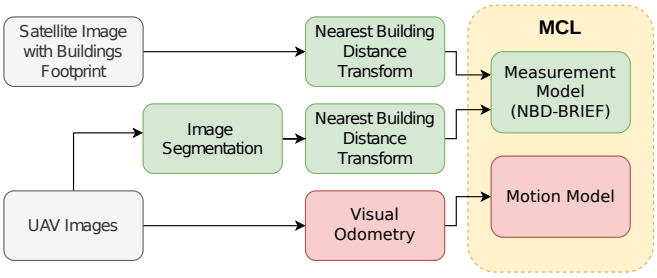


Fig. 4: Overview of the proposed framework. The distance transform is applied in the segmented UAV image and in the reference map, which is used in the MCL together with the odometry.

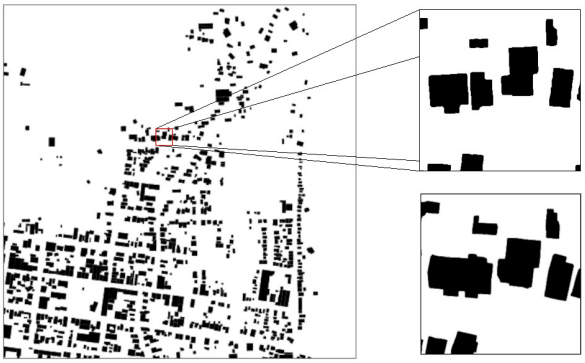


Fig. 5: Reference map, $\mathbf{M}$, used in Flight 1. The building footprint was extracted from Google Maps. Both segmented images in the right column represent the same place on Earth. The difference is that the upper image is a patch from $\mathbf{M}$, whereas the lower one is the segmented UAV image associated with the highlighted area of the map.

The reference map, $\mathbf{M}$, is a satellite image containing information about buildings, which is later converted to a map describing the proximity to buildings through the application of a distance transform. $\mathbf{M}$ could be obtained by segmenting areas associated with buildings in an RGB satellite image; however, this step is unnecessary. Nowadays, maps of building footprints are widely available in tools like Google Maps and OpenStreetMap and are incredibly detailed. In Fig. 5, we can see a satellite image with a building footprint overlay, which has been retrieved from Google Maps. The process is similar to the UAV Images; however, the images must first be segmented to extract the building information and converted to buildings proximity information.

[1]Kernels of different sizes can be used in parallel to obtain more information. In the tests, three sizes were used, as suggested in [13].

[2]The other MCL steps, such as resampling and sampling using the visual odometry as input for the motion model, follow the implementation of MCL using abBRIEF. For more details see [11].

B. Image segmentation

The first step of the proposed approach is to segment the UAV images as building or non-building, done by a framework based on the U-Net convolutional neural network [15]. Since the segmentation process is not a contribution of this work, we have chosen one of the most popular and accurate segmentation methods. U-Net comprises a contracting path and a symmetric expanding path, in which it encodes and decodes the features in a segmented image. The network we use, [16], was pre-trained using a dataset [17] composed of images from African cities and their infrastructure, building footprint, and vegetation, which are similar to the regions where our experiments were performed.

It is worth noting that even though the segmentation of buildings may not be 100% accurate, which means that the input information for the particle filter is not completely reliable, small changes in the contours of the buildings generate a small impact on the values of distances to the nearest building. Of course, if there are serious errors in the classification of buildings, or if the buildings present in the reference map change in relation to the buildings observed by the vehicle, the localization process will be harmed and may fail. But this is an inherent problem of the localization task, where observations are expected to match properly with the available map. The particle filter is generally a successful strategy when the differences between the expected and observed information are small [5], [14].

C. Computing Nearest Building Distance

Segmented images are difficult to match, as shown in Fig. 2. We propose using a modified version of the BRIEF descriptor to improve the results. It uses information about the distance to the nearest building edge, which is called ***NBD-BRIEF***.

First, the buildings' edges are extracted from the segmented image using the Canny edge detector [18], resulting in an image $\mathbf{I}$. Next, $\mathbf{I}$ is used to compute the euclidean distance of a given pixel $\mathbf{p}$ to the nearest building edge, given by

$$d(\mathbf{p}, \mathbf{I}) = \min \left\{ \min_{\mathbf{q} \in \mathbf{I}^e} \sqrt{(p_u - q_u)^2 + (p_v - q_v)^2}\ ,\ L \right\}, \quad (1)$$

where p_u and p_v are the coordinates of the pixel $\mathbf{p}$, q_u and q_v are the coordinates of an edge pixel $\mathbf{q}$ given the set of all edge pixels $\mathbf{I}^e \subseteq \mathbf{I}$, and L is a maximum distance limit. Fig. 1c presents an example of the resulting image by applying the distance transform over all pixels of image $\mathbf{I}$.

The maximum distance L is required because the UAV only observes buildings inside its current field of view; hence, an image without buildings has zero distance information. This is not the case in the reference map, where all buildings are known *a priori*. If there is no distance limit, even regions far away from buildings will have distance information in the reference map, leading to inconsistencies against the descriptors computed on UAV images. In our tests, the ideal limit for the distance transform in the UAV images lies between 25% and 50% of the biggest image dimension (length or width). The limit in the map is multiplied by an estimate of the scale between the map and the image, which depends on the height of the UAV[3].

In the NBD-BRIEF, a set of pixel pairs $\mathbf{S} = \{\mathbf{s}_1, ..., \mathbf{s}_k\}$ are randomly selected over an image, where each pair $\mathbf{s} = \{\mathbf{x_1}, \mathbf{x_2}\} \in \mathbf{S}$ is composed of two random points. During the selection, a Gaussian distribution is used in the image center to minimize the effects of buildings suddenly appearing near the edges of the image.

The binary feature vector $\mathbf{B} = (\tau_1, \tau_2, \cdots, \tau_k)$ describing the NBD-BRIEF is composed by k binary comparisons τ. Instead of comparing the intensity of two pixels, such as in abBRIEF [11], we compare the distance of these pixels to the nearest building edge, as given by

$$\tau(\mathbf{I}; \mathbf{s}) = \left\{ \begin{array}{ll} 1 & : d(\mathbf{x_1}, \mathbf{I}) < d(\mathbf{x_2}, \mathbf{I}) \\ 0 & : otherwise \end{array} \right\} \quad (2)$$

As in [11], the similarity of two images is given by calculating each NBD-BRIEF image descriptor's Hamming distance (Hd). Finally, we model the observation likelihood in the MCL with a zero-mean Gaussian distribution computed in function of such distance,

$$p(\mathbf{I}_t \mid \boldsymbol{x}_t^{[p]}, \mathbf{M}) = \mathcal{N}(Hd(\mathbf{B}_t, \hat{\mathbf{B}}_t^{[p]}), 0, \sigma), \quad (3)$$

where $\mathbf{B}_t$ is the descriptor computed from the UAV image $\mathbf{I}_t$, $\hat{\mathbf{B}}_t^{[p]}$ is the descriptor computed from the expected image obtained from $\mathbf{M}$ associated to a particle in pose $\boldsymbol{x}_t^{[p]}$, and the deviation σ is an intrinsic noise parameter of the model[4]

IV. Experiments

A. Setup

The experimental validation was made with three datasets of flights taken by two UAV models equipped with GPS, camera, and IMU in the state of Rio Grande do Sul, Brazil. Figs. 6a, 6c, and 6e show the trajectories of the flights, and further information about the flights is present in Table I.

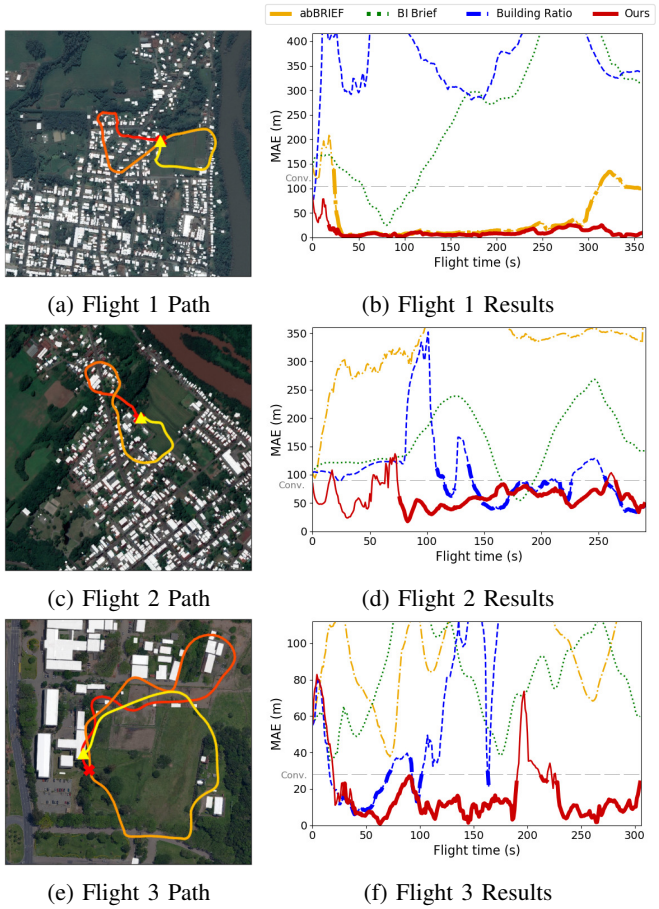
TABLE I: Flights details

	Flight 1	Flight 2	Flight 3
Location	Arroio do Meio	Arroio do Meio	Porto Alegre
UAV	DJI Phantom 3	DJI Matrice 100	DJI Matrice 100
Gimbal	Yes	No	No
Flight time	358 s	290 s	306 s
Path dist.	1800 m	1120 m	1085 m
Map area	1.09 km²	0.81 km²	0.08 km²
Altitude	35m - 130m	35m - 180m	40m - 50m

The satellite images and building footprints used in our experiments were taken from Googles Maps. We could have used any other source of satellite images, but for the region

[3]Since the particle filter used in the MCL considers maximum and minimum particle heights, we also can compute the maximum and minimum scale difference (pixels $\times$ meter) between the reference map and UAV images. During the experiments, we selected a fixed scale value from inside such interval, as shown in Section IV, which proved to work well, despite being an approximation.

[4]In our tests, we used σ as 15% of the maximum Hamming distance.



(a) Flight 1 Path (b) Flight 1 Results

(c) Flight 2 Path (d) Flight 2 Results

(e) Flight 3 Path (f) Flight 3 Results

Fig. 6: **a, c, e:** Maps and paths of the three flights used in the tests. The white parts are the building footprints overlaying a satellite image of the flight area. The flight starts on the red 'X' and ends in the yellow triangle. **b, d, f:** Comparisons of Mean Absolute Error (MAE) obtained in the three tested scenarios. The error line is drawn thicker when the particle filter converges to a single cluster. Our method kept the MAE below the convergence threshold most of the time. The abBRIEF performed well in Flight 1, while the building ratio method achieved good results in Flight 2.

where the UAVs flew, Google Maps is the one that provides the most detailed image in general. The maps are dated from 2021, and each flight was tested on only one map as older building footprints are unavailable.

All three tests present challenges. The first test uses a bigger reference map, and the third flight was recorded over an area with far fewer buildings than the other two. Besides, the second and the third flights have been recorded without a gimbal, which messes with the visual odometry[5].

In terms of comparisons with other approaches, we selected the original abBRIEF work [11]; a variation of the Building Ratio[6] approach [13]; and Binary-Input (BI) BRIEF,

[5]As an illustration, although flights 1 and 2 pass over the same region, due to the presence of a gimbal, that stabilizes the camera and facilitates image matching, the visual odometry computed with images from Flight 1 presented an error 40% smaller than that of Flight 2.

[6]The original method [13] does not use a particle filter, but a completely different location estimation framework, thus we only tested the impact of the building ratio as a measurement model. The tested framework is the same as ours, but applying the building ratio instead of the NBD-BRIEF.

TABLE II: Results of NBD-BRIEF with different distance limit

	Metric	Distance Limit L=80	L=100	L=120
Flight 1	MAE (m)	21.74	**11.26**	**11.26**
	MAE Conv. (m)	20.04	9.16	**8.74**
	Proper Conv. (%)	92.21	**95.19**	94.92
	Wrong Conv. (%)	**0**	0.02	**0**
	No conv. (%)	7.79	**4.79**	5.08
Flight 2	MAE (m)	**61.32**	61.48	62.59
	MAE Conv. (m)	59.02	**57.57**	57.75
	Proper Conv. (%)	71.35	**71.75**	71.02
	Wrong Conv. (%)	1.04	1.04	**0.90**
	No conv. (%)	27.61	**27.21**	28.08
Flight 3	MAE (m)	17.49	17.15	**15.68**
	MAE Conv. (m)	**10.00**	10.84	10.81
	Proper Conv. (%)	57.35	56.81	**58.02**
	Wrong Conv. (%)	4.97	4.34	**3.88**
	No conv. (%)	**37.68**	38.85	38.01

the same variation of abBRIEF discussed in Section II, which uses directly the segmented images, instead of the original colored ones. The abBRIEF was tested with images in the RGB color space, as in [11]. For the modified versions using Building Ratio and BI-BRIEF, we used the UAV segmented images and the reference map containing buildings information (the same used in our method before applying the distance transform).

All experiments were made with images captured at 1 FPS, 50.000 particles in the MCL, and 30 runs for each configuration, as same as done by Mantelli et al. [11]. The ground truth used for evaluation was directly obtained from the GPS coordinates during flights.

B. Results

As our work is based on MCL, the localization of the UAV is complete after the particles converge. We consider that the particles converge when the distance error is lower than 10% of the reference map size.

In the first set of tests, we evaluate our method with three different limits for the distance transform: 80, 100, and 120 pixels. Table II presents the results of Mean Absolute Error (MAE), Mean Absolute Error after convergence, and the percentages of proper, wrong, and no convergence during the length of the runs. We can see that the variation of the parameter L generally does not imply large variations in the results, which is good because it shows that the method can work with robustness keeping the parameter fixed, even knowing that the height of the UAV varies during flights. This happens because the method does not need to know the exact distance from a point to a building, which would be a measure that depends on the current scale of the image, and, consequently, the exact UAV's height. On the contrary, it is enough to know if the point is closer to the building than another point in the same image, and this relation of magnitude between the distances is maintained even if the scale is somewhat wrong (ensuring that the distance limit is sufficiently large). That said, L=100 and L=120 had a slightly better performance than L=80; thus, we fixed L as

TABLE III: Comparisons with other methods

	Metric	Methods			
		BI-BRIEF	Building Ratio	abBRIEF	Ours L=100
Flight 1	MAE (m)	263.37	372.49	39.09	**11.26**
	MAE Conv. (m)	–	–	30.75	**9.16**
	Proper Conv. (%)	0	4.49	84.96	**95.19**
	Wrong Conv. (%)	**0**	5.19	0.56	0.02
	No conv. (%)	100	90.32	14.48	**4.79**
Flight 2	MAE (m)	160.36	105.46	336.27	**61.48**
	MAE Conv. (m)	–	66.55	–	**57.57**
	Proper Conv. (%)	0	46.21	0.26	**71.75**
	Wrong Conv. (%)	**0**	3.50	0.62	1.04
	No conv. (%)	100	50.29	99.12	**27.21**
Flight 3	MAE (m)	86.15	333.00	107.13	**17.15**
	MAE Conv. (m)	–	16.26	–	**10.84**
	Proper Conv. (%)	0	16.11	4.03	**56.81**
	Wrong Conv. (%)	**0**	5.80	1.24	4.34
	No conv. (%)	100	78.09	94.73	**38.85**

100 to compare with other methods.

The comparisons of our method, BI-BRIEF, Building Ratio, and abBRIEF are shown in Table III. Our method performed better than the three other methods in all three scenarios. Our method achieved an error of 9.16 meters in the first flight, against 39.09 meters from abBRIEF, the only of the three other methods with good convergence. This was the same dataset used by Mantelli et al. in [11], so abBRIEF was expected to work well. In the second and third flights, our method had an error of 8.98 meters and 5.42 meters, respectively, much lower than the other methods.

Fig. 6 shows the MAE of the four methods in the three tested scenarios. In Flight 1, our method achieved convergence earlier than the abBRIEF method and sustained a low error along the trajectory. The abBRIEF method significantly increased the error after around 300s of the flight, where a patch of vegetation changes colors throughout the seasons.

The earliest convergence was also achieved by our method in the second flight. In this scenario, the building ratio method and ours kept the MAE below the convergence threshold most of the time, but in some areas with a low number of buildings, the error increased above the threshold.

In the third flight, the building ratio method converged before our method, but it got lost by around 100s of the flight, while ours kept the error below 20 meters most of the time. It is worth mentioning that Flight 3 is the most difficult dataset, as a large part of the flight takes place over homogeneous areas without buildings, so these sections lack information to maintain an adequate pose estimate. Even so, the NBD-BRIEF is the only method that maintains convergence most of the time.

V. Conclusion

This paper proposes a novel descriptor called NBD-BRIEF (nearest building distance BRIEF) for use in the visual UAV localization problem. We tested our proposal in challenging scenarios, with traveled distances always superior to 1km and altitudes varying from 35m to 180m – situations in which virtually no other tested approach works consistently well. Our proposed method obtained low average error throughout the experiments. In all tests, our method outperformed the competing approaches, such as the abBRIEF method, the one which inspired us to do the current work.

In the proposed strategy, we used only information about proximity to buildings, which proved to be suitable for applications in urban regions. Nonetheless, as we can see in Flight 3, the low building count in some scenarios can be a problem. In future work, we intend to investigate the use of other semantic classes, such as roads and high or low vegetation. Using more classes could improve localization and diminish the dependency on just one type of information.

References

[1] C. A. Thiels, J. M. Aho, S. P. Zietlow, and D. H. Jenkins, "Use of unmanned aerial vehicles for medical product transport," vol. 34, no. 2, 2015, pp. 104–108.

[2] F. G. Costa, J. Ueyama, T. Braun, G. Pessin, F. S. Osório, and P. A. Vargas, "The use of unmanned aerial vehicles and wireless sensor network in agricultural applications," in *2012 IEEE International Geoscience and Remote Sensing Symposium*, 2012, pp. 5045–5048.

[3] J. G. Serna, F. Vanegas, F. Gonzalez, and D. Flannery, "A review of current approaches for uav autonomous mission planning for mars biosignatures detection," in *IEEE Aerospace Conf.*, 2020, pp. 1–15.

[4] F. Caballero, L. Merino, J. Ferruz, and A. Ollero, "Improving vision-based planar motion estimation for unmanned aerial vehicles through online mosaicing," in *2006 IEEE ICRA*. IEEE, 2006, pp. 2860–2865.

[5] A. Couturier and M. A. Akhloufi, "A review on absolute visual localization for uav," *Robotics and Autonomous Systems*, vol. 135, p. 103666, 2021.

[6] A. Viswanathan, B. R. Pires, and D. Huber, "Vision-based robot localization across seasons and in remote locations," in *IEEE ICRA*. IEEE, 2016, pp. 4815–4821.

[7] G. Conte and P. Doherty, "An integrated uav navigation system based on aerial image matching," in *Aerospace Conf.* IEEE, 2008, pp. 1–10.

[8] M. Shan, F. Wang, F. Lin, Z. Gao, Y. Z. Tang, and B. M. Chen, "Google map aided visual navigation for uavs in gps-denied environment," in *2015 IEEE ROBIO*, 2015, pp. 114–119.

[9] A. L. Majdik, D. Verda, Y. Albers-Schoenberg, and D. Scaramuzza, "Air-ground matching: Appearance-based gps-denied urban localization of micro aerial vehicles," *Journal of Field Robotics*, vol. 32, no. 7, pp. 1015–1039, 2015.

[10] S. H. Choi and C. G. Park, "Image-based monte-carlo localization with information allocation logic to mitigate shadow effect," *IEEE Access*, vol. 8, pp. 213 447–213 459, 2020.

[11] M. Mantelli, D. Pittol, R. Neuland, A. Ribacki, R. Maffei, V. Jorge, E. Prestes, and M. Kolberg, "A novel measurement model based on abbrief for global localization of a uav over satellite images," *Robotics and Autonomous Systems*, vol. 112, pp. 304–319, 2019.

[12] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "Brief: Binary robust independent elementary features," in *Computer Vision – ECCV 2010*, K. Daniilidis, P. Maragos, and N. Paragios, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 778–792.

[13] J. Choi and H. Myung, "Brm localization: Uav localization in gnss-denied environments based on matching of numerical map and uav images," in *IEEE/RSJ IROS 2020*, 2020, pp. 4537–4544.

[14] S. Thrun, "Probabilistic robotics," *Commun. ACM*, vol. 45, no. 3, p. 52–57, mar 2002.

[15] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *MICCAI 2015*, N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, Eds. Cham: Springer, 2015, pp. 234–241.

[16] DrivenData, "Open cities ai challenge: Segmenting buildings for disaster resilience," https://github.com/drivendataorg/open-cities-ai-challenge/, 2020, accessed: 2021-07-29.

[17] GFDRR Labs, "Open cities ai challenge dataset, version 1.0," Radiant MLHub https://doi.org/10.34911/rdnt.f94cxb, 2020, accessed: 2021-07-29.

[18] J. Canny, "A computational approach to edge detection," *Transac. on pattern analysis and machine intelligence*, no. 6, pp. 679–698, 1986.