Hierarchical Continual Learning for Single Image Aerial Localisation

Aldrich A. Cabrera-Ponce, Manuel Martin-Ortiz, and Jose Martinez-Carranza*

Benemerita Universidad Autonoma de Puebla (BUAP), Puebla, Mexico
*Instituto Nacional de Astrofisica, Optica y Electronica (INAOE), Puebla, Mexico

ABSTRACT

Due to weather-related signal loss, GPS-based aerial localisation is challenging for Unmanned Aerial Vehicles (UAVs). Consequently, vision-based techniques have been devised to tackle this problem by leveraging the cameras integrated into UAVs. The main objective is to achieve UAV localisation throughout a flight mission using aerial images and Convolutional Neural Networks (CNNs). To address this, we introduce an aerial localisation methodology that integrates continual learning and a multi-model approach, augmented by the concept of sub-mapping inspired by Simultaneous Localisation and Mapping (SLAM) systems. This methodology involves mapping an area into different zones and re-localising the camera when it reaches a known point. We compared results using the ORB-SLAM2, a keyframe searching method based on colour histograms, and a single model to validate our methodology. Our results demonstrate that it is possible to find corresponding sub-maps and acquire the camera pose from aerial images, achieving an average accuracy of 0.77 and a processing speed of 69 fps.

1 INTRODUCTION

Aerial localisation presents a challenge for UAVs that rely on GPS coordinates for outdoor flight missions. This reliance on GPS devices often hinders pose capture and has led to the development of several vision-based methods such as feature matching [1], Visual Odometry (VO) [2], and SLAM [3]. However, these methods are computationally expensive, leading to deep-learning methods with CNNs to estimate camera poses. For the latter, CNN architecture is trained using datasets with pose labels, resulting in a learning model with sufficient accuracy to estimate the camera poses from a single image.

Current research uses the PoseNet architecture [4] to regress the camera pose using a single image. This architecture can solve the aerial localisation problem and determine the camera poses on the UAV [5]. Nevertheless, training the model can be time-consuming during flight missions due to the large dataset and limited computational memory. An alternative approach to tackle this issue is to use continual learning strategies, which enable incremental training of CNN with small dataset samples and avoid catastrophic forgetting when new information arrives.

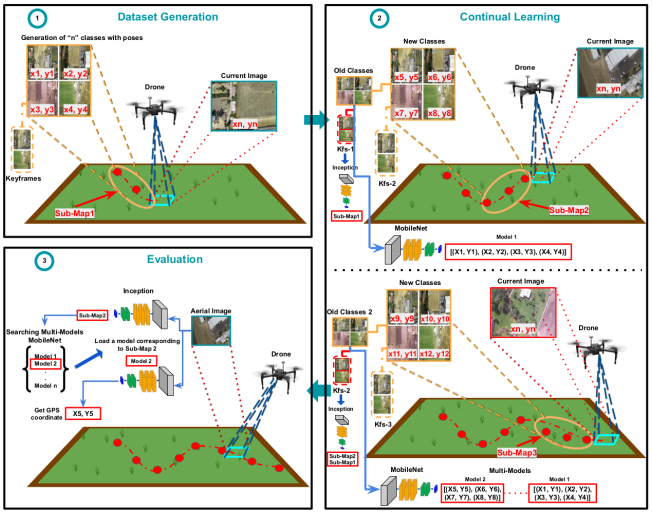

Figure 1: The outlined framework consists of three stages: 1) Dataset creation with poses designated as classes; 2) Continual learning using MobileNet for camera localisation and InceptionV4 to train the keyframes for sub-maps; 3) Multi-model evaluation, accomplished through keyframe searching from InceptionV4 results.

A latent replay strategy can be helpful in training models for in-flight missions, especially when dealing with constrained resources or uncertain surroundings. CNNs can adjust and enhance their performance without re-training, continuously updating the model with new information whilst leveraging previous knowledge. This consists of the repeatability of the last data patterns in external memory and integrating them with the new incoming data. Furthermore, the segmentation of the environment into distinct sections can yield advantages, facilitating a more concentrated and targeted approach to the learning process. Overall, continual

*Email address(es): carranza@inaoep.mx

learning strategies can improve the performance of models used, where real-time adaptability is necessary for localisation tasks.

Motivated by the above, we propose a methodology that learns hierarchically with two networks employing a latent replay strategy. To this end, we generated the sub-maps throughout the trajectory, where representative keyframes were extracted for each sub-map and trained the first network to localise the map in the area. Then, we train the second network with camera pose information in a multi-model fashion where each model generated represents one sub-map with flight coordinates information. Finally, we evaluate a testing dataset to identify the sub-map and load the corresponding model to obtain camera poses. An overview of our framework, which is divided into three stages, is presented in Figure 1.

To present our work, this paper is organised as follows. Section 2 discusses related works on localisation and continual learning approaches. Section 3 outlines the dataset generation process, sub-maps creation, and continual learning using the two architectures. Section 4 presents the experimental design and a comparison with other methods for localisation. Finally, conclusions and future work are summarised in Section 5.

2 RELATED WORK

Aerial localisation for outdoor scenarios is often challenging for UAV flight missions, especially for autonomous navigation tasks. Several works have proposed solutions to obtain aerial localisation from methods such as multi-sensor fusion with vision [6], visual odometry [7], feature matching [1], and SLAM. The latter has played an essential role in robotics, allowing us to map scenarios from features extracted in images. Therefore, one of the best-known SLAM systems is ORB-SLAM2 [3, 8] used in inspection tasks where GPS is denied [9].

In contrast, deep learning has had a significant advance using convolutional neural networks (CNNs) for place recognition tasks [10], geo-localisation with cross-view and satellite imagery [11], and goal localisation [12]. The latter uses reinforcement learning to decouple scans on different goals finding nearby localisations. However, repetitive reproducibility may delay the acquisition of the pose. Instead, other methods obtain localisation by comparing map mosaics [13], dense scenario match [14], and using multi-task networks with visual odometry, and auxiliary learning [15].

In this way, PoseNet is an architecture designed for pose estimation from an image [4]. This advantage has been used for pose estimation, and localisation under multiple scenes, sharing the images' features across the entire scenario [16]. Nonetheless, pose acquisition is still linked to end-to-end training with large datasets. An alternative is to freeze the learning in some layers inside the architecture, reducing resource consumption in the training stage [17]. On the other hand, in [5, 18], eliminate some layers in the CNN to leverage the time and parameterisation reduction with a compact network.

Due to the time and training constraints present in traditional learning, recent works have focused on continual learning strategies using small data [19, 20]. Strategies based on latent replay of data represent better performance in robotics tasks, keeping accuracy and avoiding catastrophic forgetting of prior learning [21–23]. However, these works are focused on continual learning for classification tasks, resulting in other solutions for localisation. For example, in [24, 25], they present hierarchical models for predicting semantic 2D worlds and incremental mapping from 3D measurements with stored representative features. These works allow prediction and scenario reconstruction to obtain the localisation using a single image.

On the other hand, [26] presents a method based on a buffer with 3D information of the scene and strategic samples. iMAP [27], maps an implicit scenario updating the discrepancies with keyframes to fill unsee parts. This has led to the goal of obtaining the pose using mapping and 3D modelling of the scenario, acquiring the localisation from a single image. To the same end, [28, 29] present methodologies for a continual SLAM using two architectures to learn and adapt in new scenarios and using a continual dictionary with a Quadratic Bayesian Surprise (QBS). Consequently, [30] present a sub-mapping approach with a multi-model process and continuous learning for aerial localisation.

Finally, these works give us a panorama where localisation is performed from sub-maps, multi-scenes, and multi-models using a hierarchical scheme. Motivated by the latter, in this paper, we propose continual hierarchical learning and adopt the SLAM concept to learn small parts of the scenario as a way of sub-mapping, thus creating learning models for each sub-map. Regarding SLAM, our approach can be seen as a hierarchical system that identifies keyframes to load a model corresponding to the sub-map and obtain the image's pose.

3 METHODOLOGY

This section outlines the methodology for continual hierarchical learning with two deep network architectures: MobileNetV1 and InceptionV4. Firstly, we collected aerial images from a monocular camera onboard the UAV and associated them with GPS coordinates during the flight. Afterwards, we save a keyframe set for each trajectory section, which we consider sub-maps of the whole path. Finally, we incrementally trained the networks to create a model containing the pose information and another model containing the submap information.

3.1 Dataset Generation

We split the dataset generation process into two steps. The first step involved obtaining aerial images paired with GPS coordinates, while the second entailed creating

keyframes to represent sub-maps. For the latter, the Robot Operating System (ROS) facilitated communication with the drone, and the Ground Control Station (GCS) received the image stream and corresponding GPS data. We perform initial flight missions to capture images at a resolution of 128×128 pixels and convert GPS coordinates into meters. Thus, we generate small samples of flight coordinates divided into classes, yielding about 50 classes for each trajectory. In addition, we defined sub-maps containing the information of 5 flight coordinates and created a new map when the drone covers a distance of 50 meters. Finally, we save keyframes for each submap to train the InceptionV4 network, whose information determines the trajectory zone to which they belong.

Figure 2 illustrates the creation and division of sub-maps with 5 flight coordinates and the storage keyframes with three images for each sub-map. These keyframes represent the images of the sub-maps beginning, middle and end, corresponding to classes 1, 3, and 5, respectively. Besides, we consider the distance between the points to create a new sub-map when the drone travels approximately 50 to 100 metres.

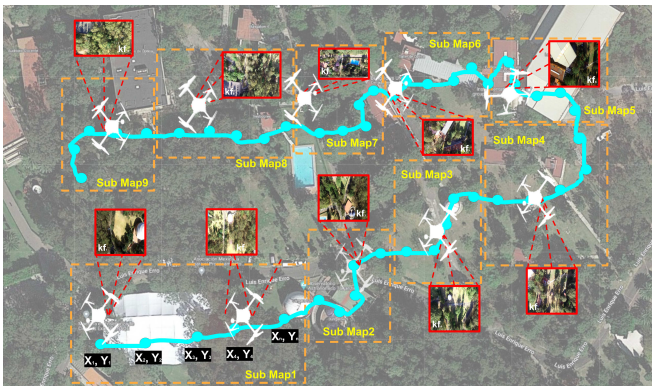

Figure 2: Training dataset acquisition, sub-maps creation, and keyframe saving. Cian shows the flight dron trajectory, where the circles represent GPS coordinates. The orange rectangles illustrate the sub-maps created.

In Figure 3, we visually represent the trajectories overlaid in Google Earth using GPS-derived information. Simultaneously, Figure 4 illustrates the covered area of the routes transformed into meters. These trajectories represent the flights carried out to collect the training dataset and generate the keyframes, and the flight coordinates are presented by red circles in Figure 4. The length of the traversed trajectory is 0.53 km for trajectory 1, 1.4 km for the second, and 2.4 and 2.9 km for trajectories 3 and 4, respectively. Moreover, we have 47 flight coordinates for trajectory 1, 50 for the second and third trajectories, and 52 for the last trajectory, representing the classes along the entire path.

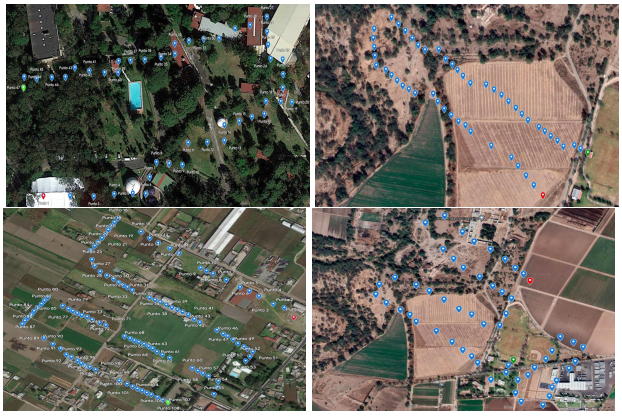

Figure 3: Flight paths into Google Earth with GPS information like waypoints. We show the beginning path in a red waypoint and the end in green.

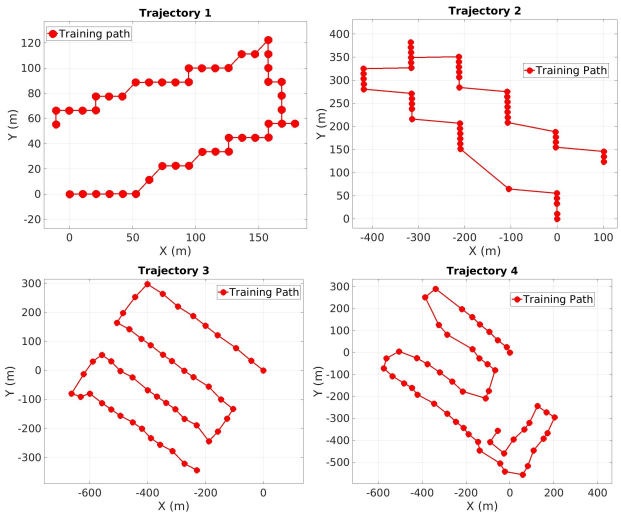

Figure 4: GPS coordinates in metres with a total trajectory length of 1) 0.53 km with 42 poses; 2) 1.4 km with 50 poses; 3) 2.4 km with 50 poses; 4) 2.9 km with 52 poses.

3.2 *Continual Learning*

For this section, we adopt the concept of sub-mapping used in SLAM systems, which divides the whole trajectory into several parts to achieve better data processing and establish aerial localisation under a hierarchical learning methodology. The goal is to acquire the image's poses once it is located within one of the sub-maps along the trajectory. To achieve this, we split the methodology into two parts and continually train two CNNs: MobileNetV1 and InceptionV4.

First, we use the continual learning strategy called latent replay to train MobileNetV1. This allows us to train on the fly information, saving the essential patterns in external memory, where each pattern set represents the classes. This method combines new data with old ones, rejuvenating the previously

learned weights and consolidating learning. As a result, the network can learn 1000 classes whose previous information is stored in a latent layer within an external memory in the *pool6* layer. However, we set a limit of continuous learning with around 50 classes to avoid saturating the memory with parameters and to prevent catastrophic forgetting.

In this way, the network continuously learns the 50 poses it receives from the information obtained from the drone, collecting 200 images for each established coordinate. Splitting the labels into subsections can improve the evaluation step, training a model with five coordinates representing one sub-map in the entire path. To train the InceptionV4, we save three keyframes of the sub-maps generated, as shown in Figure 5. The flow of information is trained with MobileNetV1, and the keyframes are stored for training the InceptionV4 network. Finally, we keep 27 keyframes for trajectory 1, 30 for trajectories 2 and 3, and 52 for trajectory 4.

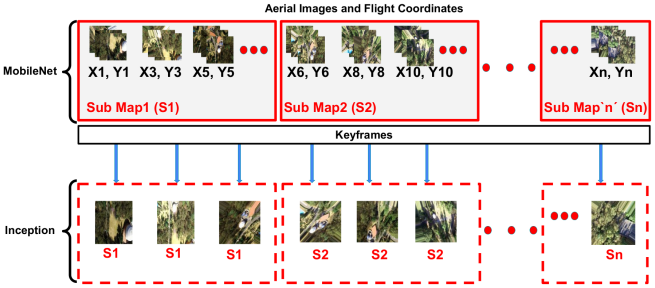

Figure 5: Continual training of the networks in a hierarchical way where MobileNet is trained using aerial images with information on flight coordinates and InceptionV4 with keyframes generated for each sub-map as labels.

Secondly, we employ continual learning of InceptionV4, which uses the keyframes and their labels to select the sub-map for the aerial image. The network is structured to learn an input image, whereby we encapsulate the features within a vector format. As a new keyframe of the matching class appears, we update the weights within a temporal vector and combine them with the previous ones. Conversely, if a different keyframe class arrive, we assign new features in a new index, expanding the features vector to adjust information from both classes. At the end of the training, we concatenate the features vector with the temporal vector, merging all keyframe features into one. This approach enables the network to learn from the continuous data stream and update its weights dynamically, improving its performance over time.

Thereby, hierarchically, we have a methodology capable of localising an aerial image inside the trajectory and acquiring the position. To carry out the continual learning of the networks, we take the following parameters summarised in Table 1. It is all on one computer with Cuda 11.1, PyTorch 1.9, and 8GB RAM with a GeForce 640M Nvidia card.

Table 1: Parameters used for continual training of the networks MobileNetV1 and InceptionV4.

Network	MobileNet	InceptionV4
Optimiser	SGD	SGD
Batch size	128	1
Epoch	20	1
Loss Function	Cross Entropy	Cross Entropy
Learning Rate	0.001	0.001
Latent layer	Pool6 Layer	Temporal Vector
Patterns/Features	3500	1536

4 EXPERIMENTS AND RESULTS

We carried out two experiments to assess the efficiency of our hierarchical aerial localisation approach. The first experiment aimed to identify the corresponding sub-map for the given aerial image using the InceptionV4 and a colour histogram-based method. In the second experiment, we leveraged the hierarchical framework to obtain the poses of an input image and ascertain the associated sub-map labels. We then loaded the MobileNet model with pose information to establish localisation. We also evaluated a single model, the ORB-SLAM2 re-localisation module, and a keyframe search using the colour histogram to compare our approach. Finally, we measured each method's performance speed expressed in frames per second (fps), running on Ubuntu 20.04, Python 3.8, OpenCV 3 and ROS Noetic.

4.1 Sub-Maps Results

In this experiment, we trained the InceptionV4 network on the fly using previously saved keyframes generated while creating sub-maps. Each keyframe is labelled with an index representing the corresponding sub-map number. When a new keyframe of a different class arrives, the network extracts the features and stores their weights in a temporary vector, merged with the previous ones. In this way, once the final sub-map is trained, the feature vector contains information on all keyframes learned, along with their respective labels. To evaluate the learning model, we conducted 4 test flights near the training trajectory, where each image was fed as the input to the network. The network's output provides the sub-map index and the total number of features corresponding to the closest keyframe. Thus, the network returned the label with the highest similarity of features found in the stored keyframes, which corresponded to the sub-map class.

We have implemented a keyframe search engine based on the image's colour to provide a comparison. Firstly, we divide the keyframe into five quadrants and extract the colour histogram, which is then concatenated into a single histogram. This provides more precision in finding descriptors during the evaluation step. Subsequently, we calculate the colour histogram of each image and compare it with the keyframes using the chi-square metric, where a value close to zero indicates a higher similarity with the keyframe. Tables 2 and

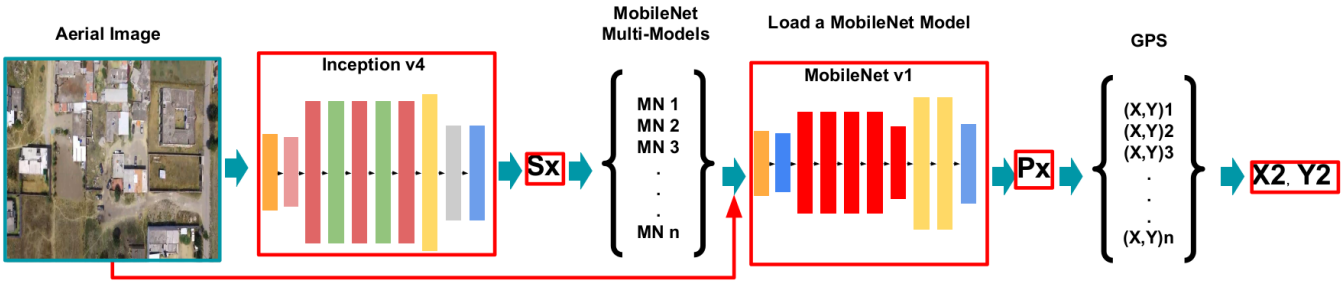

Figure 6: Hierarchical aerial localisation. We hierarchically evaluate each image, first passing for the InceptionV4 network to find the corresponding sub-map. Then, given the sub-map, we load the MobileNet model with information on 5 flight coordinates. Finally, the same image passes through the model to find the camera pose and get the aerial localisation using a single image.

Table 2: Accuracy results with the test dataset using a colour histogram descriptor to find the submap.

Trajectory	SubMap	Kfs Found	Accuracy
1	9	87	0.6041
2	10	65	0.5555
3	10	56	0.6666
4	10	310	0.5107

Table 3: Accuracy results with the test dataset using the InceptionV4 network to find the submap.

Trajectory	SubMaps	Kfs Found	Accuracy
1	9	117	0.8125
2	10	84	0.7179
3	10	60	0.7142
4	10	467	0.7693

3 present the result of the first experiment involving InceptionV4 and colour histogram. We also report the number of correctly localised images, their corresponding keyframes, and accuracy results. This experiment provides an overview of the keyframe search, highlighting the effectiveness of the InceptionV4 network in learning trajectory zones, given its continual training.

4.2 Hierarchical Learning Results

The second experiment evaluates our approach using the hierarchical concept for aerial localisation. For this, we follow the diagram in Figure 6, where we pass an input image through the IncetionV4 network, resulting in the sub-map label to which it corresponds. Then, we load the MobileNet model of the sub-map found and evaluate the input image, giving in the output one of the five classes learned. We argue that methodology can get the zone localisation and image's pose in a level fashion in contrast to the traditional localisation frameworks.

For comparison, we evaluate other methods to visualise aerial localisation using a single learning model, the ORB-SLAM2 localisation module, and a keyframe search engine using the colour histogram. The learning of a single model consists of continual learning with the latent replay strategy of the data. However, instead of having one model for each sub-map, we learn and update all classes in a single model. This can impact learning, leading to catastrophic forgetting of previous data when more information arrives on the network.

On the one hand, in the ORB-SLAM2 re-localisation module, we create the map using the training dataset and save the image poses in a text file. Then, we use the test dataset, deactivating the mapping to re-localise the images with feature matching. Thus, a test image with the same descriptors is automatically re-localised inside the map. Nevertheless, we take the distance between the test frame and the keyframe to recover the position by determining the coordinate corresponding to the nearby keyframe. On the other hand, we apply our same hierarchical methodology for the histogram results, changing the Inception network for the colour histogram, and we evaluate the test dataset. For this case, we define the bins of Hue, Saturation and Value channels in (8, 12, 3) to have a better result in the searching step.

Hence, we present in Table 4 the comparison results for an aerial localisation task. In the table, we expose the information on the poses inside the entire path, the number of testing images, and the accuracy of each method. This accuracy consists of the number of correctly re-localised images from acquiring the sub-map to the corresponding pose. Our methodology keeps the best localisation results, successfully obtaining more images re-localised on the first two trajectories with an average accuracy of 0.74. In comparison, ORB-SLAM2 performs better on the last two trajectories with an average accuracy of 0.81.

This evaluation indicates that our methodology wins in the first two trajectories while the SLAM system obtains more

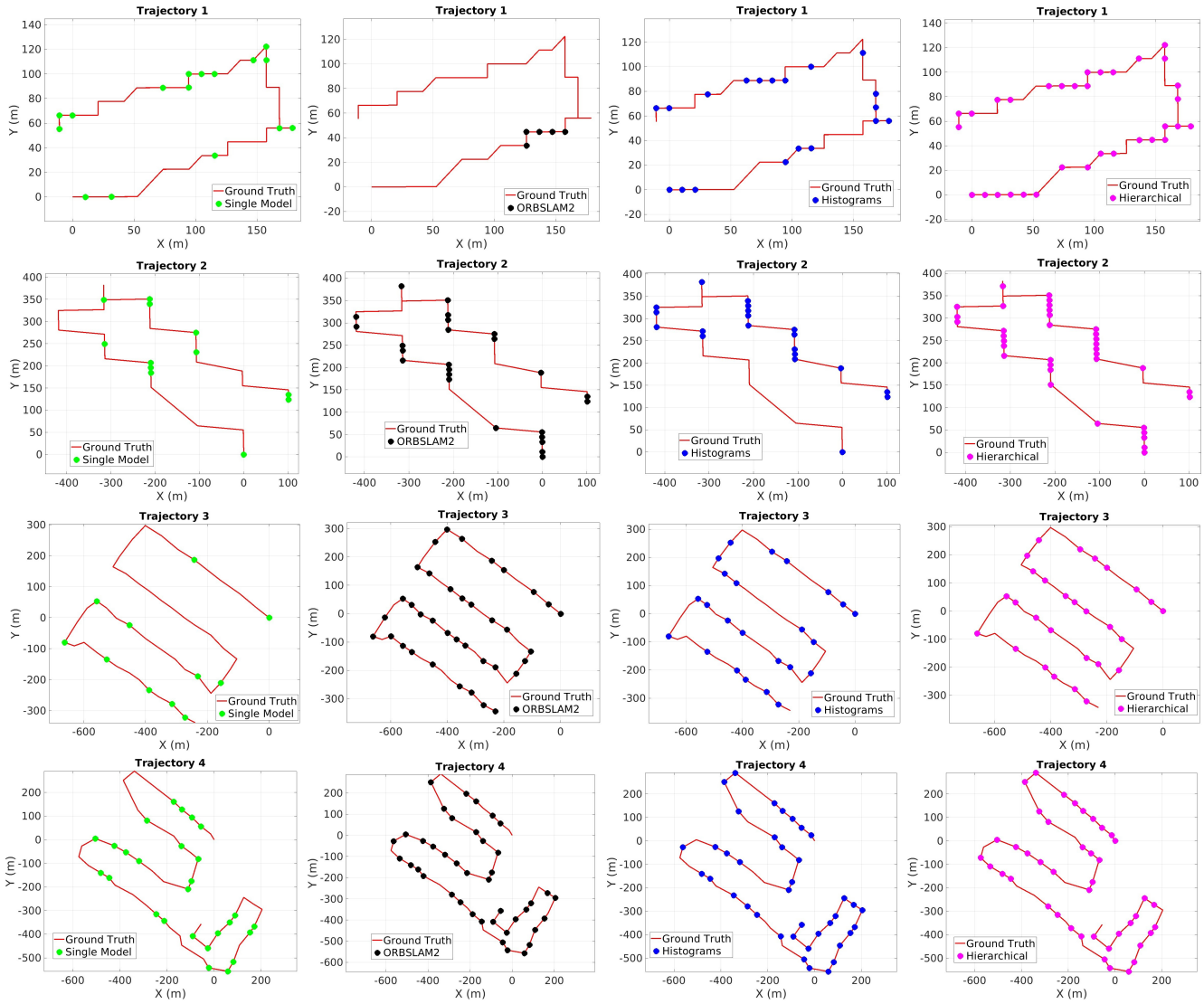

Figure 7: We utilised test images to re-localise and establish the ground truth based on the training trajectory. The circles in the results represent the recovered poses for each evaluated method. The first column corresponds to the evaluation using a single model, the second with ORB-SLAM2, the third using a histogram colour method, and the last with our hierarchical approach.

poses in the last two. Nevertheless, we present a similar result to SLAM, arguing that localisation is achieved in these trajectories. For the latter, the SLAM system drops its performance in complicated scenarios lacking descriptors or with a repetitive pattern, such as trajectories 1 and 2. In contrast, a deep learning methodology keeps the accuracy enough to establish the localisation, learning the essential features. Nonetheless, the results with a single model demonstrate our assumptions regarding catastrophic forgetting of the first data. Even with a continual learning strategy, the network loses knowledge as more information arrives.

The results of the learning method with a single model demonstrate a maximum accuracy of 0.43. Conversely, the methodology using a keyframe search system with a colour histogram improves pose acquisition to below 0.67. Nevertheless, the similarity of several images in colour affects this outcome, particularly in the first two trajectories. Thus, a colour histogram and feature extraction may not be advantageous, particularly in complex trajectories. Therefore, a multi-model-based system has the potential to alleviate the burden of rigid and data-sensitive training by enabling the learning of poses to be split into different models.

As a final result, we present the camera poses recovered with each comparison method using a testing trajectory similar to the training. Thus, if an aerial image is re-localised, we obtain the sub-map and its pose close to the ground truth. Figure 7 shows the trajectories results and poses returned using each method. We can see that in some cases, there are empty

Table 4: Accuracy results for re-localisation poses using comparison methods.

Trajectory	Poses	Images	Single Model	Histrograms	ORB-SLAM2	Hierarchical
1	47	144	0.2500	0.3055	0.1041	0.7083
2	50	117	0.2564	0.3076	0.5897	0.7777
3	50	84	0.2976	0.6071	0.9166	0.8928
4	52	607	0.4382	0.6690	0.7166	0.7001

Table 5: Fps results using the comparison methods.

Approach	Traj. 1	Traj. 2	Traj. 3	Traj. 4
Single model	55.30	48.28	55.64	62.76
Histogram	59.63	62.40	64.80	61.87
ORB-SLAM2	85.47	83.33	92.57	89.28
Hierarchical	77.44	68.52	67.63	63.37

spaces in the trajectories, and this is because the method can't obtain the correct localisation in that zone. In addition, to analyse the response time of each method, we present in Table 5 the processing speed in fps, where ORB-SLAM2 obtains a higher speed but not so far from that obtained with our approach.

Finally, our method outperforms the others when considering the number of correctly localised images, as shown in Table 6. However, ORB-SLAM2 demonstrates superior re-localisation performance for trajectories 3 and 4 but not for the earlier ones. We argue that our methodology could be helpful for inspection tasks and as a backup localisation method in case the GPS signal is lost, as it obtains a greater number of images re-located even in the last trajectories.

Table 6: Comparison of the images used to retrieve the poses by each method in the four flight trajectories.

Approach	Traj. 1	Traj. 2	Traj. 3	Traj. 4
Single model	36	30	25	225
Histogram	44	36	51	266
ORB-SLAM2	15	69	**77**	**435**
Hierarchical	**102**	**91**	75	425

5 CONCLUSION

We have presented a hierarchical continual learning approach to aerial localisation using two networks and a concept based on sub-mapping. In addition, we have developed a multi-model process for each sub-map divided into the entire trajectory, with information on the flight coordinates. The methodology aims to identify a sub-map that best represents the course using the InceptionV4 network to determine the image location and load the corresponding model to obtain the pose. To achieve this, we have continuously trained the networks during a flight mission using the latent replay strategy while storing representative keyframes of the sub-map to determine which model to load. As a result, this methodology offers the advantage of maintaining good accuracy while reducing the computational resources in the training step.

In addition, to alleviate catastrophic forgetting, we proposed dividing the training into multiple training models, each with information on five camera poses. Our approach outperforms the localisation results compared to a single model training and a methodology based on the colour histogram. Nonetheless, we maintain enough accuracy with the ORB-SLAM2 system. Moreover, we have demonstrated this approach as a backup localisation for UAVs with an average accuracy of 0.77 and a performance speed of 69 fps, which is sufficient for real-time systems. For future work, we will improve the methodology to inspire new techniques based on autonomous navigation, localisation and vision tasks using continual learning strategies, including a regression network to estimate poses instead of getting the label class.

ACKNOWLEDGEMENTS

The first author is thankful for his scholarship funded by Consejo Nacional de Humanidades, Ciencia y Tecnología (CONAHCYT) under grant 802791.

REFERENCES

[1] Peter Hansen, Peter Corke, and Wageeh Boles. Wide-angle visual feature matching for outdoor localization. *The International Journal of Robotics Research*, 29(2-3):267–297, 2010.

[2] Ramon Gonzalez, Francisco Rodriguez, Jose Luis Guzman, Cedric Pradalier, and Roland Siegwart. Combined visual odometry and visual compass for off-road mobile robots localization. *Robotica*, 30(6):865–878, 2012.

[3] Raul Mur-Artal and Juan D Tardós. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE transactions on robotics*, 33(5):1255–1262, 2017.

[4] Alex Kendall, Matthew Grimes, and Roberto Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization. In *Proceedings of the IEEE international conference on computer vision*, pages 2938–2946, 2015.

[5] Aldrich A Cabrera-Ponce and J Martinez-Carranza. Aerial geo-localisation for mavs using posenet. In *2019 Workshop on Research, Education and Development of Unmanned Aerial Systems (RED UAS)*, pages 192–198. IEEE, 2019.

[6] Ghasem Abdi, Farhad Samadzadegan, and Franz Kurz. Pose estimation of unmanned aerial vehicles based on a vision-aided multi-sensor fusion. In *XXII ISPRS Congress, Technical Commission I*, volume 41, pages 193–199, 2016.

[7] Ruben Mascaro, Lucas Teixeira, Timo Hinzmann, Roland Siegwart, and Margarita Chli. Gomsf: Graph-optimization based multi-sensor fusion for robust uav pose estimation. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1421–1428. IEEE, 2018.

[8] Carlos Campos, Richard Elvira, Juan J Gómez Rodríguez, José MM Montiel, and Juan D Tardós. Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam. *IEEE Transactions on Robotics*, 37(6):1874–1890, 2021.

[9] Diego Benjumea, Alfonso Alcántara, Agustin Ramos, Arturo Torres-Gonzalez, Pedro Sánchez-Cuevas, Jesus Capitan, Guillermo Heredia, and Anibal Ollero. Localization system for lightweight unmanned aerial vehicles in inspection tasks. *Sensors*, 21(17):5937, 2021.

[10] Andrea Vallone, Frederik Warburg, Hans Hansen, Søren Hauberg, and Javier Civera. Danish airs and grounds: A dataset for aerial-to-street-level place recognition and localization. *IEEE Robotics and Automation Letters*, 7(4):9207–9214, 2022.

[11] Akshay Shetty and Grace Xingxin Gao. Uav pose estimation using cross-view geolocalization with satellite imagery. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 1827–1833. IEEE, 2019.

[12] Aleksis Pirinen, Anton Samuelsson, John Backsund, and Kalle Åström. Aerial view goal localization with reinforcement learning. *arXiv preprint arXiv:2209.03694*, 2022.

[13] Noe Samano, Mengjie Zhou, and Andrew Calway. Global aerial localisation using image and map embeddings. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5788–5794. IEEE, 2021.

[14] Shitao Tang, Chengzhou Tang, Rui Huang, Siyu Zhu, and Ping Tan. Learning camera localization via dense scene matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1831–1841, 2021.

[15] Abhinav Valada, Noha Radwan, and Wolfram Burgard. Deep auxiliary learning for visual localization and odometry. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 6939–6946. IEEE, 2018.

[16] Hunter Blanton, Connor Greenwell, Scott Workman, and Nathan Jacobs. Extending absolute pose regression to multiple scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 38–39, 2020.

[17] MS Müller, S Urban, and B Jutzi. Squeezeposenet: Image based pose regression with small convolutional neural networks for real time uas navigation. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 4:49, 2017.

[18] Aldrich A Cabrera-Ponce and Jose Martinez-Carranza. Convolutional neural networks for geo-localisation with a single aerial image. *Journal of Real-Time Image Processing*, 19(3):565–575, 2022.

[19] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2001–2010, 2017.

[20] Xiaofan Yu, Yunhui Guo, Sicun Gao, and Tajana Rosing. Scale: Online self-supervised lifelong learning without prior knowledge. *arXiv preprint arXiv:2208.11266*, 2022.

[21] Kevin Thandiackal, Tiziano Portenier, Andrea Giovannini, Maria Gabrani, and Orcun Goksel. Generative feature-driven image replay for continual learning. *arXiv preprint arXiv:2106.05350*, 2021.

[22] Vincenzo Lomonaco and Davide Maltoni. Core50: a new dataset and benchmark for continuous object recognition. In *Conference on Robot Learning*, pages 17–26. PMLR, 2017.

[23] Lorenzo Pellegrini, Gabriele Graffieti, Vincenzo Lomonaco, and Davide Maltoni. Latent replay for real-time continual learning. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10203–10209. IEEE, 2020.

[24] Robin Karlsson, Alexander Carballo, Keisuke Fujii, Kento Ohtani, and Kazuya Takeda. Predictive world models from real-world partial observations. *arXiv preprint arXiv:2301.04783*, 2023.

[25] Xingguang Zhong, Yue Pan, Jens Behley, and Cyrill Stachniss. Shine-mapping: Large-scale 3d mapping using sparse hierarchical implicit neural representations. *arXiv preprint arXiv:2210.02299*, 2022.

[26] Shuzhe Wang, Zakaria Laskar, Iaroslav Melekhov, Xiaotian Li, and Juho Kannala. Continual learning for image-based camera localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3252–3262, 2021.

[27] Edgar Sucar, Shikun Liu, Joseph Ortiz, and Andrew J Davison. imap: Implicit mapping and positioning in real-time. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6229–6238, 2021.

[28] Niclas Vödisch, Daniele Cattaneo, Wolfram Burgard, and Abhinav Valada. Continual slam: Beyond lifelong simultaneous localization and mapping through continual learning. *arXiv preprint arXiv:2203.01578*, 2022.

[29] Ali Safa, Tim Verbelen, Ilja Ocket, André Bourdoux, Hichem Sahli, Francky Catthoor, and Georges Gielen. Learning to slam on the fly in unknown environments: A continual learning approach for drones in visually ambiguous scenes. *arXiv preprint arXiv:2208.12997*, 2022.

[30] Aldrich Alfredo Cabrera-Ponce, Manuel Isidro Martin-Ortiz, and Jose Martinez-Carranza. Multi-model continual learning for camera localisation from aerial images. In G. de Croon and C. De Wagter, editors, *13^{th} International Micro Air Vehicle Conference*, pages 103–109, Delft, the Netherlands, Sep 2022. Paper no. IMAV2022-12.