2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

OrthoSwarm: Orthoimagery Drone Swarms

Tuhao Zhao[1,2], Peng Yi[2,3], Haozhou Zhai[1,2], and Tianjiang Hu[1,2,3*]

***Abstract*— This paper addresses the urgent need for rapid synthesis of georeferenced orthoimages in post-disaster scenarios, where pre-disaster satellite maps cannot be directly reused due to significant urban changes. Drone swarms offer advantages of large scale, wide aerial view and rapid coverage, of disaster-stricken areas. However, synthesizing georeferenced orthoimages within limited time remains challenging without camera calibration, primarily due to inevitable inconsistencies in intrinsics and extrinsics across different cameras, as well as sensor errors. To tackle this issue, we propose OrthoSwarm, a parallelizable calibration-free system architecture that leverages drone swarms rectilinear path planning and pre-disaster satellite maps for efficient orthoimage synthesis. OrthoSwarm's performance is validated on a self-constructed benchmark dataset, generated by drone swarms in a digital twin city covering 3 natural disaster scenarios(debris, waterlogging, haze), with real-world validation using real single-drone aerial videos split into segments to simulate swarm acquisition. Experimental results from both simulated and real-captured data confirm the effectiveness of the proposed approach, enabling fast and visually consistent georeferenced orthoimage synthesis in stable post-disaster environments to support first responders promptly.**

I. INTRODUCTION

Drone swarms enable efficient imagery acquisition over large disaster areas, which can be processed and fused into georeferenced orthoimages within a limited time. This allows first responders to quickly gain full situational awareness and optimize rescue strategies, reducing casualties and property losses in post-disaster scenarios.

Undoubtedly, heterogeneous drone swarms enable faster large-scale disaster-area imagery acquisition than single drones. Nevertheless, rapid georeferenced orthoimage generation faces two key challenges. The first is high-frame-rate processing under memory constraints: among state-of-the-art methods, Map2DFusion [1] is one of the fastest but only achieves around 25 FPS, limiting it to small-scale swarms. Most existing methods also suffer from memory limits, as they store all input images and related data in memory, restricting them to small-area orthoimage generation. The second challenge involves camera intrinsics requirements: some mosaic-based methods [2] depend on pre-known camera intrinsics, while SLAM-based [3], [4] and deep learning methods [5]–[7] can estimate intrinsics automatically but require far more time than rescue operations allow. Moreover, acquiring intrinsics for large-scale heterogeneous swarms (especially those with zoom lenses) is difficult and time-consuming, so such reliance greatly limits algorithm applicability.

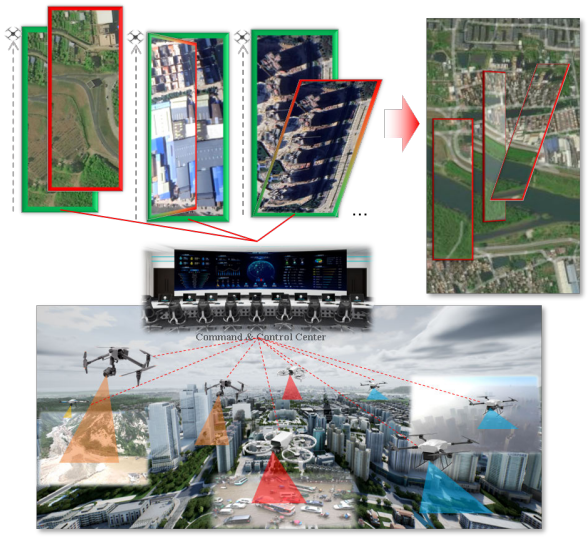


Fig. 1. Drone swarms are widely employed in the search-and-rescue tasks within disaster-related scenarios, while common mis-matching phenomena often occur in drone image geolocation and orientation, and erroneous synthesized orthoimages. From left to right in the upper-left corner, three common errors are: inconsistency between actual and sensor-derived image geolocation, inconsistency between actual and sensor-derived image orientation, and time-accumulated geolocation error caused by sensor anomalies.

To address these challenges, this paper proposes a calibration-free and parallelizable system for georeferenced orthoimage generation in post-disaster emergency mapping. A core strategy to reduce computational cost is minimizing redundant computations: given that camera extrinsics(geolocation and orientation), drone polyline trajectories, and disaster-unaffected regions are consistent with pre-disaster satellite maps, re-computing these data is computationally inefficient. Thus, our system leverages readily available prior information and integrates a real-time georeferenced orthoimage generation module and two coarse-to-fine refinement modules, both with strong locality to enable parallel acceleration. While the real-time preliminary orthoimage does not affect the final result, it provides a quick rough overview to improve user experience. An overview of the system is shown in Fig.2. Our main contributions are

* This work was jointly Supported by Guangdong S&T Program (Key-Area Research and Development Program of Guangdong Province) with Granted No.2024B1111060004 and the National Natural Science Foundation of China under Grant 62473390. (Corresponding author: Tianjiang Hu, hutj3@mail.sysu.edu.cn)

[1] School of Artificial Intelligence, Sun Yat-sen University, Guangzhou 510275, China

[2] Zhuhai Key Laboratory on Collective Intelligence and Unmanned Systems, Zhuhai 519082, China

[3] School of Aeronautics and Astronautics, Sun Yat-sen University(Shenzhen Campus), Shenzhen 518106, China

All the emails are correspondingly {zhaoth5, zhaihzh, yipeng3}@mail2.sysu.edu.cn and hutj3@mail.sysu.edu.cn.

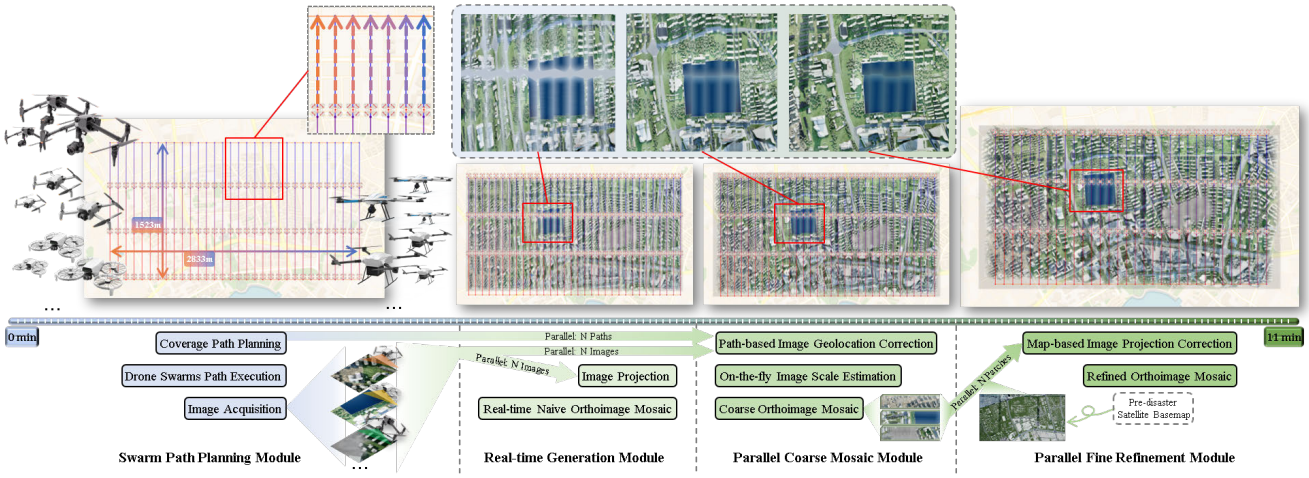


Fig. 2. Overview of the proposed system architecture, which comprises four modules. First, the swarm path planning module handles path planning and execution, while acquiring imagery from each drone during flight. Next, the real-time generation module and parallel coarse mosaic module both taking N images as input and processing each image in independent processes, output a full orthoimage and coarse-refined orthoimage patches, respectively. Finally, the parallel fine refinement module performs parallel fine refinement on each patch and outputs the fully refined orthoimage. For a $6.35km^2$ area, the entire procedure takes only ~11 minutes.

summarized as follows:

- **Drone swarms-oriented system**: A modular system integrating path planning, image acquisition, and orthoimage generation with refinement is proposed for drone swarms applications, significantly reducing computational overhead through cross-module collaboration and parallel processing.
- **Pre-disaster satellite map-based and calibration-free orthoimage generation method**: A method supports parallel acceleration, enabling scalability for large-scale drone swarms in generating orthoimages of extensive disaster-stricken areas while exhibiting reliable performance in weak-texture regions under stable conditions.
- **Disaster-prone-city Scenario Dataset**: A dataset is constructed in a $6.35km^2$ digital twin city, covering three natural disaster scenarios (debris flow, urban waterlogging, haze), with images acquired by a heterogeneous drone swarms (96 drones equipped with 5 types of cameras).

II. RELATED WORKS

Extensive research has been devoted to orthoimage synthesis. In the early stages, mosaic-based methods dominated, however, they are limited by low accuracy and a narrow coverage area. Subsequently, SLAM-based methods emerged, enabling large-area orthoimage generation with higher accuracy but introducing substantial computational overhead. Recently, efforts have been made to integrate these two approaches, yielding promising results.

A. Mosaic-based Methods

Mosaic-based methods [8]–[10] typically follows a 5-step pipeline: acquisition, preprocessing, ortho-rectification, registration, and fusion. The acquisition phase ensures sufficient image overlap for subsequent processing. Preprocessing involves denoising and preliminary geolocation correction. Ortho-rectification employs pinhole-based models(e.g., collinear equations [11], affine transform [12], and polynomial refinement [13]) combined with interpolation techniques or neural networks like WDSR [14] for hole filling. However, neural networks remain underutilized due to high computational demands and marginal performance gains. Image registration remains a critical research focus, with sparse feature matching methods(e.g., SuperPoint [15], LightGlue [16]) and dense feature matching methods(e.g., RoMA [17]) demonstrating varying trade-offs between speed and accuracy. Cross-modal matching approaches(e.g., MINIMA [18]) present promising alternatives for mosaic-based workflows. The fusion stage typically employs weighted blending [8] to address overlapping regions.

B. SfM-based and SLAM-based Methods

SfM-based and SLAM-based methods [1], [19]–[22] achieve high-accuracy orthoimage synthesis by leveraging 3D environment models. These approaches reconstruct 3D scenes using SfM algorithms [23], [24] or VSLAM algorithms [3], [4], and subsequently synthesize georeferenced orthoimages by integrating the 3D model as a digital elevation model(DEM) with camera intrinsics and extrinsics. While 3D models enable diverse outputs (e.g., point clouds, DSMs), orthoimage synthesis remains a computationally intensive task with limited practical value given the marginal gains in accuracy compared to mosaic-based alternatives.

C. Combination Methods

Recent studies have explored combined approaches integrating mosaic-based and SLAM-based methods to achieve accurate and rapid orthoimage synthesis for small drone swarms [25], [26]. However, these methods rely on feature extraction and matching from all acquired images, leading to computational bottlenecks under high-concurrency scenarios(e.g., middle or large drone swarms). To address this limitation, we propose a parallel acceleration system

for georeferenced orthoimage generation in large area using large-scale drone swarms.

III. METHODOLOGY

The 4 main modules of our system and their steps are illustrated in Fig.2. After determining the target area, the swarm path planning module uses Boustrophedon Cellular Decomposition(BCD) [27] or other methods to generate polyline coverage paths for each drone for stable post-disaster mapping. During flight, images are captured at a fixed rate, and camera extrinsics such as geolocation and orientation are recorded simultaneously. In the real-time generation module, an instantaneous orthoimage is produced using only captured images and camera extrinsics for quick situational awareness. Meanwhile, the parallel coarse mosaic module starts processing with additional trajectory data. It estimates image scaling factors using image features and camera extrinsics, and performs preliminary localization correction by aligning image positions with pre-planned paths. Images with similar scale and extrinsics are grouped to generate multiple coarse orthoimage patches. When a drone finishes its path, the parallel fine refinement module fuses these patches with pre-disaster satellite basemaps for geometric correction. A transformation matrix is estimated to align coarse patches with the satellite reference; for disaster areas with large scene changes, robust matches from undamaged regions compensate for local mismatches and adjust image geolocations. The final georeferenced orthoimage with consistent visual quality is generated using corrected images in position, orientation, and scale. Implementation details of these modules are elaborated in this section.

A. Swarm Path Planning Module

To enable orthoimage generation in disaster-stricken areas, drone swarms requires coverage paths to ensure complete spatial coverage of target areas. While coverage paths also play an important role in path-based image geolocation correction. Specifically, images captured with similar camera orientations are gathered together for georeferencing refinement, where curved trajectories significantly complicate this process. Furthermore, curved paths inherently increase the path execution time due to trajectory complexity, making polyline-based coverage paths generated by algorithms such as BCD [27], TMSTC* [28] or column-by-column the preferred choice. Besides, since our system have no idea about the camera intrinsics of the drone swarms, in order to avoid missing areas, the field of view(FoV) of each camera in the path planning algorithm is replaced with $\hat{\theta}$(a value less than the possible minimum FoV).

During path execution, high-frequency image and related geolocations and orientations acquisition is strategically prioritized despite the system's reliance on geolocation-based mosaicking rather than feature-based matching. While the absence of feature extraction and matching requirements eliminates the need for strict overlap constraints between sequential images, increased image density directly enhances the orthogonality of the final orthoimage through more detailed sampling. Notably, this approach maintains computational efficiency since the system avoids feature extraction image-by-image, as elaborated in subsequent algorithmic analyses and validated in experimental evaluations.

B. Real-time Generation Module

To address the computational challenges of synthesizing orthoimages from high-frequency image streams(hundreds of images per second), we adopt a simplified synthesis approach based on planar hypothesis. This method leverages a FoV-constrained frustum camera model, which incorporates both the geolocation and orientation parameters of each image along with the FoV $\hat{\theta}$(the same parameter used in coverage path planning algorithm). Then, the homography matrix H mapping the image plane to the georeferenced orthoimage can be computed by establishing a correspondence between the 4 corner coordinates of the image and their respective geolocations. Subsequently, each image is projected onto the orthoimage using this homography matrix.

The overlapping regions in the projected image space is a critical factor in achieving high-fidelity orthoimage synthesis. In an ideal scenario, the final orthoimage should be composed of the most orthogonal pixels, corresponding to image regions where the viewing direction is orthogonal to the ground plane. To formalize it, we introduce a Euclidean distance-based metric that quantifies the orthogonality of each pixel $\boldsymbol{P}$ in the projected image relative to the orthogonal center $\boldsymbol{P}_{oc}$ as follow:

$$Orthogonality(P_{oc}, P) = \frac{1}{\|\boldsymbol{P}_{oc} - \boldsymbol{P}\|_2} \tag{1}$$

The global orthogonality of the synthesized orthoimage can be formally formulated as the sum of each pixel's orthogonality scores. To maximize the global orthogonality, we utilize a weighted replacement strategy where each pixel of the orthoimage is assigned the pixel value from the corresponding projected image that achieves the maximum orthogonality score. Other approaches such as weighted average blending are also commonly employed in orthoimage generation, they inherently leading to perceptual blurring and loss of fine spatial details. In contrast, our approach maintains sharp feature boundaries and more details, and it is significant for users and our following modules which reuse this step.

C. Parallel Coarse Mosaic Module

The coarse mosaic module operates in parallel with the real-time generation process to preliminarily correct image geolocation errors and estimate image scaling factors. Drone imagery geolocation errors primarily stem from stochastic perturbations induced by mechanical vibrations or sensor measurement uncertainties, which manifest as small-magnitude, spatially correlated deviations along the drone trajectory. To mitigate these errors, we leverage the polyline-based flight paths generated by the drone swarms control module(as detailed in Sec.III-A) to establish a geometric reference for geolocation correction. Given that drone trajectories consist of straight-line segments(denoted as $S =$

$\{s_1, s_2, ...s_n\}$), we partition the trajectory into m segments and associate each image with its corresponding segment based on image orientation. For each segment s_i, we model the expected geolocation distribution as a linear function $y = kx + b$, where the parameters k and b are derived via least-squares regression of the observed image geolocations $\{(x_1, y_1), (x_2, y_2), ...(x_n, y_n)\}$:

$$k = \frac{n\sum_{i=1}^{n} x_i y_i - \left(\sum_{i=1}^{n} x_i\right)\left(\sum_{i=1}^{n} y_i\right)}{n\sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2}$$
$$b = \frac{1}{n}\left(\sum_{i=1}^{n} y_i - k\sum_{i=1}^{n} x_i\right) \tag{2}$$

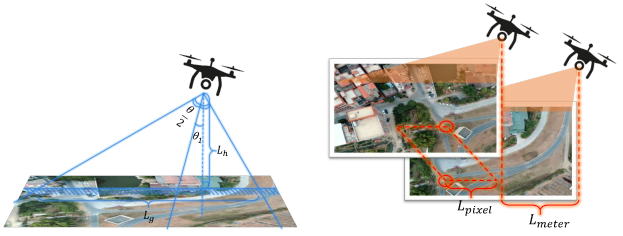
The corrected geolocation $(\hat{x}_i, \hat{y}_i)$ for each image is then computed by projecting the original coordinates onto the fitted line:

$$\hat{x}_i = \frac{x_i + k(y_i - b)}{1 + k^2}$$
$$\hat{y}_i = \frac{kx_i + k^2 y_i + b}{1 + k^2} \tag{3}$$

By aligning image geolocations with the polyline trajectory, we obtain coarse but robust image geolocations. For image scale estimation, we derive the geographic scale from the camera FoV and drone altitude. Assuming the ground plane at zero altitude, the FoV θ is calculated using the geometric model in Fig.3 and the following equation:

$$\theta = 2 * (\theta_1 + arctan(\frac{\frac{L_g}{2} - L_h * tan(\theta_1))}{L_h})$$
$$L_g = \frac{L_{pixel}}{L_{meter}} * W_{image} \tag{4}$$

where θ_1 means the pitch of the camera, L_h, L_g are drone



(a) FoV estimation model (b) L_g estimation model

Fig. 3. Graphical illustration geometric relationships in 4, where FoV is derived from drone altitude and camera orientation, and the ground-visible range(in meter). The ground-visible range L_g is determined by the Euclidean distance between matched feature points in overlapping images and the geodesic distance of the drone's trajectory during imaging.

altitude and ground plane width in the image. While θ_1 and L_h can be directly derived from geolocation and orientation data, the estimation of L_g requires determining the scale between image pixels and real-world meters. This is achieved by first calculating the pixel-based ground width L_{pixel} using feature correspondence analysis. Specifically, a LightGlue [16] deep learning model is employed to identify matched feature points between image pairs. The L_{pixel} is then computed by measuring the Euclidean distance between feature points spatially close to the images' orthogonal center P_{oc}. To increase robustness, these feature-based measurements are further refined through a Gaussian outlier suppression model that filters noisy correspondences. The final L_g is obtained by applying this scale factor to the pixel width of the image and the FoV θ is also got.

After the estimation of FoV, the image scale is implicitly determined via the FoV-constrained frustum camera model. However, this FoV assumes a flat ground plane at zero altitude. While DEMs could improve FoV accuracy, their unavailability in post-disaster scenarios necessitates additional fine-grained refinement in subsequent processes.

D. Parallel Fine Refinement Module

The fine refinement module addresses large-scale geolocation and orientation errors that persist after coarse mosaic, which primarily eliminates random local perturbations. As illustrated in the upper-left corner of Fig.1, common large-scale errors include geolocation deviation, orientation deviations, and cumulative drift errors. These errors are particularly challenging to correct when they originate early in the mission or manifest in low-texture regions such as waterlogged or haze-affected areas. Notably, such fixed errors remain invariant until drone heading or camera orientation changes, which means these errors can be corrected in batch. Thus, we first group images with similar orientations to synthesize orthoimage patches using the coarse-refined geolocation, scale, and the synthesis method from the real-time generation module. This approach enables simultaneous correction of large-scale errors across multiple patches, significantly improving computational efficiency compared to per-image adjustments. Crucially, initial image geolocation and scale data are excluded from this process, as severe errors in these parameters can introduce significant texture duplication or loss, thereby compromising feature matching consistency during fine refinement in the next step.

Rather than employing iterative optimization frameworks like bundle adjustment for SLAM-based methods, our approach leverages satellite maps as an accessible and reliable reference for single-step error correction. This strategy not only simplifies correction but also preserves perfect locality, enabling efficient parallel processing and reducing computational cost. However, both synthesized orthoimage patches and satellite maps contain non-orthogonal distortions, leading to discrepancies between the actual geolocation of features (e.g., rooftop or tower tops) and their positions inferred from satellite maps via pixel coordinates. These discrepancies introduce erroneous references that must be carefully removed. We observe that tall buildings show height-dependent offsets on satellite maps due to elevation differences, whereas low buildings and roads remain highly consistent with little to no offset. This coherence allows our method to estimate projection matrices by preserving regions with consistent height or offset. Specifically, we use the dense feature matching algorithm RoMA [17] to obtain uniformly distributed correspondences between orthoimage patches and satellite maps. For post-disaster scenarios with severe scene changes, our method further compensates for

mismatches using robust matches from undamaged regions with stable features. A homography matrix is then robustly estimated via RANSAC to simultaneously correct translation, rotation, and scale discrepancies. This methodology ensures geometric alignment while reducing the impact of non-orthogonal distortions using reliable feature correspondences, and delivers stable georeferencing even for disaster-hit areas with large scene changes.

The final full orthoimage is synthesized by integrating all images using the fine-refined geolocation parameters, optimized scale factors, and the synthesis method established in the real-time generation module. This process corrects texture inconsistencies or lost that persist in the coarse-refined orthoimage patches, thereby achieving a globally consistent orthoimage of the target area.

E. Parallel Execution for drone swarms

As is mentioned in Sec.III-D, since the fine refinement module exclusively consumes orthoimage patches and satellite maps without modifying the latter, the only critical section are image groups corresponding to the same orthoimage patches. These groups, already formed during the coarse mosaic stage, enable independent refinement across parallel processes. This design allows for maximal parallelization each group can be assigned to a separate computational thread or process, with no inter-process synchronization required for distinct groups. The theoretical lower bound of the total execution time is thus determined by two components: the maximum processing time among all parallel threads or processes(t_{p_i}), which reflects the longest-running refinement task, and the sequential post-processing phase(t_{q_i}) required to aggregate the fine-refined patches into the final orthoimage. This formulation is formally expressed as below:

$$t = (max_{i=1}^{n}(t_{p_i}) + \sum_{i=1}^{n}(t_{q_i})) \tag{5}$$

This mathematical characterization rigorously quantifies the scalability of the system under ideal parallel computing conditions.

IV. EXPERIMENTS

To validate the effectiveness and practicality of our system, we conduct comprehensive evaluations across three key dimensions: synthesized orthoimage quality, orthoimage synthesis computational efficiency, and system scalability. Notably, the performance advantages of our system can only be fully demonstrated when evaluated on sufficiently large-scale datasets. To this end, we first construct a specialized disaster-prone-city scenario dataset for systematic testing, then compare our system against two widely used commercial software tools: Pix4DMapper [29] and PTGui. Additionally, we further assess the system on a real-world dataset to verify its robustness in practical scenarios. All experiments are performed on a workstation equipped with an Intel i9-10980XE CPU and an NVIDIA RTX 3090 GPU. Experimental results show that our system achieves the shortest orthoimage synthesis time while maintaining orthoimage quality comparable to that of the aforementioned commercial software.

A. Disaster-prone-city Scenario Dataset

The infrequency of natural disasters, particularly in urban environments, severely hinders the development of real-world datasets for disaster response research. To address this limitation, we construct a digital twin of a real-world urban environment in Unreal Engine 4, simulating diverse disaster scenarios, including debris flows, waterlogging, and haze, each covering areas of $400m \times 880m$, $330m \times 320m$, and $440m \times 190m$, respectively. These disaster-affected regions exceed the coverage of individual images and present extreme challenges for georeferenced orthoimage generation due to weak texture regions(e.g., waterlogged areas with near-zero textures) and dynamic textures(e.g., flowing smog in haze areas). As illustrated in Fig.4, the pre-disaster map(Fig.4(a)) serves as the reference satellite map for all experiments.

(a) Pre-disaster scenarios (b) Post-disaster scenarios

Fig. 4. Pre-disaster and post-disaster metropolitan scenarios are simulated in the digital twin, where the pre-disaster scenario serves as the reference satellite map. Red rectangles highlight three disaster-affected areas(from left to right: debris flow, waterlogging, haze).

To efficiently cover the $6.35km^2$ affected area, we deploy a swarm of 96 simulated drones, each equipped with one of five distinct camera configurations characterized by randomized FoVs from $\{105°, 100°, 95°, 90°, 85°\}$. The cameras maintain fixed yaw angles $[-30°, 30°]$ and pitch angles $[5°, 5°]$ throughout flight. Images are captured at a simulated altitude of $150m$ with a resolution of 640×480 pixels, and each camera's geolocation and orientation are recorded during acquisition. Notably, the dataset provides noise-free ground-truth of each camera's geolocation and orientation, enabling systematic evaluation of algorithm robustness against synthetic noise profiles. In our experiments, we introduce random positional offsets $[-1m, 1m]$ and cumulative GPS drift errors $t \times error_c$, with $error_c \in [-0.1m, 0.1m]$ to simulate realistic sensor imperfections. The drone flight paths are visualized in Fig.2.

B. Comparisons to Pre-existing Systems

We conduct a comprehensive comparison between our system and two state-of-the-art commercial orthoimage synthesis tools Pix4DMapper [29] and PTGui on the disaster-prone-city scenario dataset. Due to the dataset's large-scale nature($6.35km^2$, 32544 images), neither Pix4DMapper nor PTGui could generate the full georeferenced orthoimage within 24 hours, necessitating evaluation on a 1017-image

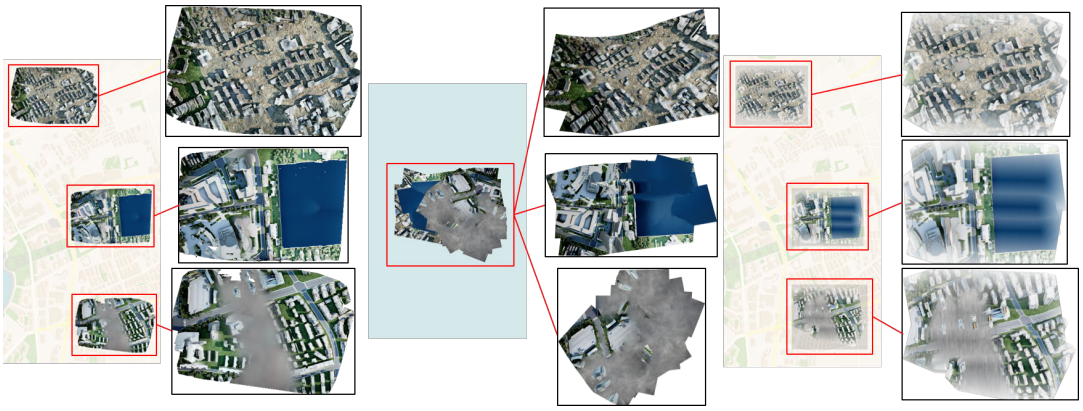


Fig. 5. Comparison of orthoimages for debris flow(upper), waterlogging(middle), and haze(lower) regions generated by Pix4DMapper, PTGui, and our system. PTGui failed to generate valid georeferenced outputs in all scenarios, with significant performance degradation observed in waterlogging and haze areas. Our system synthesizes georeferenced orthoimages with comparable visual consistency to Pix4DMapper across all disaster-affected regions.

Fig. 6. The complete georeferenced orthoimage(middle image) is synthesized from all 32544 images, with red dashed lines overlaid on roads to show consistent alignment between the background map and the orthoimage, confirming reliable georeferencing with visual consistency. The left, upper right, and lower right images present detailed views of disaster-affected areas, demonstrating clear building edges in the debris flow region, well-delineated waterlogging boundaries, and continuous road networks in the haze-affected area despite environmental obscuration.

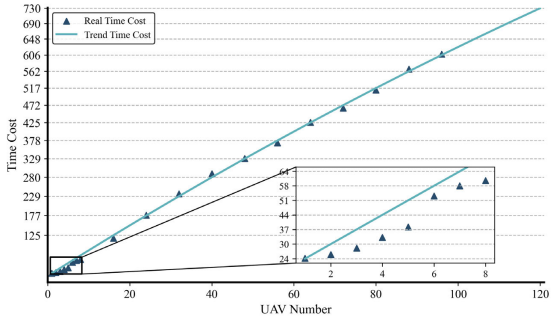


Fig. 7. This line chart illustrates the relationship between the number of drones(339 images per drone) and the time consumed for orthoimage mosaicking. Real time-cost is marked as scatter points, and the curve is a fitted from the scatter points. The detailed view shows the time-costs when the number of drones is less than 8 and computing resources are not fully loaded.

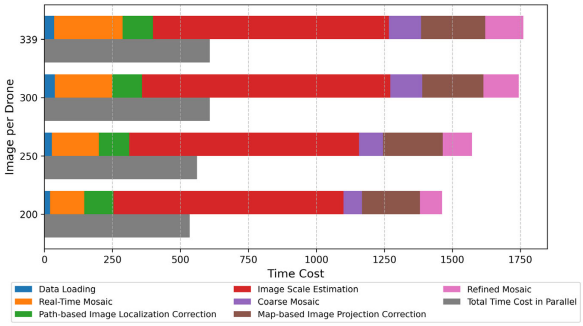


Fig. 8. This stacked bar chart illustrates the relationship between the number of images per drone and the consumed for orthoimage mosaic. The stacked bar chart above shows the total time consumed by each step of the algorithm across all sub-processes, while the bar chart below represents the total time taken for the algorithm to run in parallel.

subset. Quantitative results in Tab.I and qualitative comparisons in Fig.5 and Fig.6 demonstrate the distinct advantages of our system. In the debris flow area(Fig.5 upper), all methods successfully generated orthoimages, but only our system and Pix4DMapper produced georeferenced outputs. For the texture-deficient waterlogging region(Fig.5 middle), PTGui failed in areas near waterlogging due to its reliance on feature matching without leveraging image geolocation. While waterlogging areas occupy nearly the entire image footprint, leaving insufficient texture features for reliable matches, our system mitigates this limitation by operating on orthoimage patches with expanded FoV, preserving sufficient contextual features to ensure reliable georeferencing with visual consistency. Similarly, in the dynamic haze region(Fig.5 lower), PTGui's performance degraded due to unstable texture patterns caused by smog movement, whereas our method maintained stable visual consistency in georeferenced orthoimages through patch-based processing.

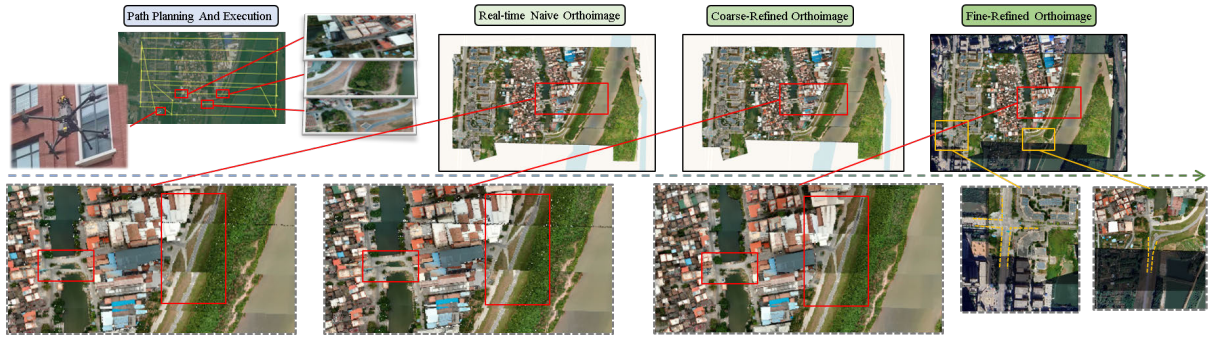


Fig. 9. Georeferenced orthoimage synthesized from imagery captured by a DJI M300 RTK equipped with a Z30 camera. In the upper row, from left to right are as follows: the DJI M300 RTK, the planned flight path of the drone, the raw aerial images acquired by the drone, the orthoimage generated by the real-time generation module, the coarse-refined orthoimage, and the fine-refined orthoimage. In the lower row, the left three images present the bridge and road features undergoing step-by-step correction; the rightmost exhibits the continuous road alignment between the background satellite map and our synthesized orthoimage. These visual results collectively demonstrate the capability of our system to generate geographically useful orthoimages in real-world physical scenarios.

TABLE I

COMPARISON BETWEEN OUR SYSTEM AND PIX4DMAPPER, PTGUI

Method / Test	Pix4D	PTGui	Ours
Is Georeferenced Orthoimage	**Y**	N	**Y**
Fully Synthesis Success	N	N	**Y**
Fully Synthesis Time	-	-	**667s**
Debris flows Mosaic Success	**Y**	**Y**	**Y**
Debris flows Mosaic Time	3399s	13263s	**36s**
Waterlogging Mosaic Success	**Y**	N	**Y**
Waterlogging Mosaic Time	7348s	12759s	**29s**
Haze Mosaic Success	**Y**	N	**Y**
Haze Mosaic Time	1073s	6078s	**29s**

The full orthoimage synthesis results(Fig.6) further highlight our system's scalability in simulated scenarios. While Pix4DMapper and PTGui exceeded practical time limits during full-scale processing, our method completed the task in 667 seconds. This efficiency stems from our optimized orthoimage synthesis strategy, which limits feature extraction and matching to a fixed number of operations per image, in contrast to the time-consuming bundle adjustment required by commercial tools. As shown in Fig.6, the red dashed lines on roads show consistent alignment between the background map and the generated orthoimage, confirming reliable georeferencing with visual consistency. Notably, our system preserves clear structural details in all disaster-affected areas: building edges remain sharp in debris flow zones, waterlogging boundaries are well delineated, and road continuity is maintained despite heavy smog obscuration.

C. Scalability Analysis

We evaluate the scalability of our system through two critical dimensions: the size of the drone swarms and the image count per drone. These metrics reflect the system's capacity to handle large-scale distributed systems and its efficiency in processing high-volume image data. The results are analyzed in Fig.7 and Fig.8. As shown in Fig.7, our system exhibits linear scalability in drone swarms size, with time cost increasing proportionally when the number of drones exceeds the CPU and memory constraints threshold(8 drones). This linear relationship confirms the system's compatibility with large-scale swarm operations. Notably, smaller swarm sizes demonstrate significantly lower time costs and higher acceleration ratios, indicating that distributed deployment across multiple computing nodes could further reduce processing time, as evidenced by the deviation from the theoretical time curve in the detailed view.

In the image count analysis(Fig.8), time cost increases with the number of images per drone(from 200 to 339 images). The stacked bar chart reveals that disk I/O-bound operations(data loading, real-time mosaic, coarse mosaic, refined mosaic) account for inherently difficult-to-optimize time consumption, as these steps are constrained by sequential memory access rather than computational parallelism. In contrast, computation-intensive modules (path-based image localization correction, image scale estimation, map-based projection correction) demonstrate parallelization potential, enabling overlapping execution with I/O operations. This explains the near-identical time costs for 300-image and 339-image scenarios, highlighting the importance of optimized scheduling to minimize idle resource cycles.

D. Physical Experiments and Discussions

We conducted a physical experiment in Foshan City using a single DJI M300 RTK equipped with a Z30 camera, with flight data recorded as video during the flight. Our system was evaluated on this dataset, which required extracting individual images from the video stream along with their geolocation and orientation data. To achieve this, we developed a dedicated script to decode images from the video and extract geospatial metadata from the video's Supplemental Enhancement Information(SEI) stream, ensuring alignment with our system's input requirements.

The synthesized orthoimages are presented in Fig.9. The flight path demonstrates reliable performance for polyline-based trajectories in stable conditions, including both generated coverage paths and manually designated routes. Notably, the real-time orthoimage and coarse-refined orthoimage exhibit minimal visual discrepancies, attributed to the pre-assumed FoV similar to the camera's actual FoV and small altitude variation in the target area. While the coarse-refined orthoimage contains visible defects, the fine-refinement module, leveraging pre-disaster satellite maps for geometric

correction, effectively eliminates these defects. The resulting fine-refined orthoimage achieves spatial alignment with satellite road networks, demonstrating reliable georeferencing with visual consistency to support emergency response operations. This validation confirms our system's capability to produce visually consistent reference maps for real-world post-disaster mapping in stable conditions.

V. CONCLUDING REMARKS

This paper has proposed and developed OrthoSwarm–a parallelizable calibration-free system architecture for georeferenced orthoimage synthesis using drone swarms with rectilinear trajectories, to support first responders promptly in post-disaster scenarios. We also construct a benchmark dataset of simulated drone swarms imagery in a digital twin city, which covers three typical natural disaster scenarios. Validations on both simulated data and real single-drone aerial videos confirm OrthoSwarm's effectiveness under its designed operating conditions.

While our system verifies the feasibility of fast calibration-free georeferenced orthoimage generation, the work can be further advanced with the following improvements. It assumes near-planar terrain, fixed altitude and polyline paths, making it unsuitable for curved trajectories or strong winds; it also lacks real-world swarm tests, and hardware restricts full parallelization in large-scale deployment. Future work will focus on addressing these limitations to enable scalable, real-time geospatial analysis for disaster response applications.

REFERENCES

[1] S. Bu, Y. Zhao, G. Wan, and Z. Liu, "Map2dfusion: Real-time incremental uav image mosaicing based on monocular slam," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 4564–4571.

[2] J. Brauchle, M. Geßner, T. Kraft, D. Hein, M. Lesmeister, J. Gonschorek, M. Bock, and R. Berger, "Regional rapid mapping for first responders-turkey 2023 earthquake," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 48, pp. 15–19, 2024.

[3] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "Orb-slam: A versatile and accurate monocular slam system," *IEEE transactions on robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[4] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. Montiel, and J. D. Tardós, "Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam," *IEEE transactions on robotics*, vol. 37, no. 6, pp. 1874–1890, 2021.

[5] A. L. Rodríguez-Santiago, J. A. Arias-Aguilar, H. Takemura, and A. E. Petrilli-Barcelo, "High-resolution reconstructions of aerial images based on deep learning," *Computación y Sistemas*, vol. 25, no. 4, pp. 739–749, 2021.

[6] X. Wang, W. Zhang, H. Xie, H. Ai, Q. Yuan, and Z. Zhan, "Torthogaussian: Splatting true digital orthophoto maps," *arXiv preprint arXiv:2411.19594*, 2024.

[7] Q. Wang, Z. Zhan, J. He, Z. Tu, X. Zhu, and J. Yuan, "High-quality spatial reconstruction and orthoimage generation using efficient 2d gaussian splatting," *arXiv preprint arXiv:2503.19703*, 2025.

[8] A. M. Pinto, H. Pinto, and A. C. Matos, "A mosaicking approach for visual mapping of large-scale environments," in *2016 International Conference on Autonomous Robot Systems and Competitions (ICARSC)*. IEEE, 2016, pp. 87–93.

[9] C.-H. Tsai and Y.-C. Lin, "An accelerated image matching technique for uav orthoimage registration," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 128, pp. 130–145, 2017.

[10] J. Zhang, S. Xu, Y. Zhao, J. Sun, S. Xu, and X. Zhang, "Aerial orthoimage generation for uav remote sensing," *Information Fusion*, vol. 89, pp. 91–120, 2023.

[11] T. Toutin, "Geometric processing of remote sensing images: models, algorithms and methods," *International journal of remote sensing*, vol. 25, no. 10, pp. 1893–1924, 2004.

[12] S. Hattori, T. Ono, C. Fraser, and H. Hasegawa, "Orientation of high-resolution satellite images based on affine projection," *International Archives of Photogrammetry and Remote Sensing*, vol. 33, no. B3/1; PART 3, pp. 359–366, 2000.

[13] T. Ono, S. Hattori, H. Hasegawa, and S.-i. Akamatsu, "Digital mapping using high resolution satellite imagery based on 2d affine projection model," *International Archives of Photogrammetry and Remote Sensing*, vol. 33, no. B3/2; PART 3, pp. 672–677, 2000.

[14] S. V. V. K. Samudrala, Y. Zhao, and R. R. Vatsavai, "Novel deep learning framework for imputing holes in orthorectified vhr images," in *IGARSS 2023-2023 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 2023, pp. 5158–5161.

[15] 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 (CVPR) Workshops*, June 2018.

[16] P. Lindenberger, P.-E. Sarlin, and M. Pollefeys, "Lightglue: Local feature matching at light speed," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 17 627–17 638.

[17] J. Edstedt, Q. Sun, G. Bökman, M. Wadenbäck, and M. Felsberg, "Roma: Robust dense feature matching," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 19 790–19 800.

[18] J. Ren, X. Jiang, Z. Li, D. Liang, X. Zhou, and X. Bai, "Minima: Modality invariant image matching," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 23 059–23 068.

[19] W. Wang, Y. Zhao, P. Han, P. Zhao, and S. Bu, "Terrainfusion: Real-time digital surface model reconstruction based on monocular slam," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 7895–7902.

[20] L. Chen, Y. Zhao, S. Xu, S. Bu, P. Han, and G. Wan, "Densefusion: Large-scale online dense pointcloud and dsm mapping for uavs," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 4766–4773.

[21] J. Pan, L. Chen, and L. Zhang, "A multi-image mosaic method for farmland uav aerial images via reference image optimization," in *2021 International Symposium on Computer Technology and Information Science (ISCTIS)*. IEEE, 2021, pp. 411–417.

[22] Y. Liu, A. Akbar, T. Yu, Y. Yu, Y. Kong, J. Gao, H. Wang, Y. Li, H. Zhao, and C. Liu, "Artemis: A real-time efficient ortho-mapping and thematic identification system for uav-based rapid response," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 229, pp. 396–421, 2025.

[23] J. L. Schonberger and J.-M. Frahm, "Structure-from-motion revisited," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4104–4113.

[24] L. Pan, D. Barath, M. Pollefeys, and J. L. Schönberger, "Global Structure-from-Motion Revisited," in *European Conference on Computer Vision (ECCV)*, 2024.

[25] J. J. Ruiz, F. Caballero, and L. Merino, "Mgraph: A multigraph homography method to generate incremental mosaics in real-time from uav swarms," *Ieee Robotics and Automation Letters*, vol. 3, no. 4, pp. 2838–2845, 2018.

[26] G. Gao, M. Yuan, Z. Ma, J. Gu, W. Meng, S. Xu, and X. Zhang, "Georos: Georeferenced real-time orthophoto stitching with unmanned aerial vehicle," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 2250–2256.

[27] H. Choset and P. Pignon, "Coverage path planning: The boustrophedon cellular decomposition," in *Field and service robotics*. Springer, 1998, pp. 203–209.

[28] J. Lu, B. Zeng, J. Tang, T. L. Lam, and J. Wen, "Tmstc*: A path planning algorithm for minimizing turns in multi-robot coverage," *IEEE Robotics and Automation Letters*, vol. 8, no. 8, pp. 5275–5282, 2023.

[29] J. Vallet, F. Panissod, C. Strecha, and M. Tracol, "Photogrammetric performance of an ultra light weight swinglet" uav"," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 38, pp. 253–258, 2012.