Digital Object Identifier

A Cross-Platform HD Dataset and a Two-Step Framework for Robust Aerial Image Matching

MD. SHAHID[1], ABHISHEK B.[2], AND SUMOHANA S. CHANNAPPAYYA[3], (Senior Member, IEEE)

[1]Department of Electrical Engineering, Indian Institute of Technology Hyderabad, Kandi 502284, India (e-mail: ee15resch02005@iith.ac.in)
[2]Department of Electrical Engineering, Indian Institute of Technology Hyderabad, Kandi 502284, India (e-mail: ee17btech11004@iith.ac.in)
[3]Department of Electrical Engineering, Indian Institute of Technology Hyderabad, Kandi 502284, India (e-mail: sumohana@ee.iith.ac.in)

Corresponding author: Md. Shahid (e-mail:ee15resch02005@iith.ac.in).

ABSTRACT Image matching has been an active research area in the computer vision community over the past decades. Significant advances in image matching algorithms have attracted attention from many emerging applications. However, aerial image matching remains demanding due to the variety of airborne platforms and onboard electro-optic sensors, long operational ranges, limited datasets and resources, and constrained operating environments. We present two contributions in this work to overcome these challenges: a) an upgraded cross-platform image dataset built over images taken from an aircraft and satellite and b) a two-step cross-platform image matching framework. Our dataset considers several practical scenarios in cross-platform matching and semantic segmentation. The first step in our two-step matching framework performs coarse-matching using a lightweight convolutional neural network (CNN) with help from aircraft instantaneous parameters. In the second step, we fine-tune standard off-the-shelf image matching algorithms by exploiting *spectral*, *temporal* and *flow* features followed by cluster analysis. We validate our proposed matching framework over our dataset, two publicly available aerial cross-platform datasets, and a derived dataset using various standard evaluation methodologies. Specifically, we show that both steps in our proposed two-step framework help to improve the matching performance in the cross-platform image matching scenario.

I. INTRODUCTION

Remotely-piloted aircraft systems (RPAS) are evolving rapidly for commercial applications due to relaxations from regulatory authorities. Remotely-piloted aircraft include Unmanned Aerial Vehicles (UAV) and drones in general. The role of UAVs has extended from reconnaissance and surveillance purposes to remote sensing, search and rescue (SAR) operations, combat missions, and so on. UAVs are classified into tactical, medium-altitude long-endurance (MALE), and high altitude long-endurance (HALE) categories. However, drones are relatively lightweight and capable of flying at low altitudes and for a short duration. UAVs and drones are divided operationally in terms of Visual Line of Sight (VLOS), Extended Visual Line of Sight (EVLOS), and Beyond Visual Line of Sight (BVLOS). The latter two typically need assistance from a radio link or satellite terminal for navigation. Navigation sensors are of utmost importance for long-range UAVs. However, navigation sensors suffer from drift issues depending upon the type or class of sensors. This drift could lead to a deviation from the intended path that could have catastrophic consequences during long endurance flights at high speeds. This drift impact is contained with Global Positioning System (GPS) input at regular intervals. A typical navigation strategy for path correction is to use GPS input at regular intervals.

However, GPS signals become unavailable or unreliable due to electromagnetic interference, atmospheric effects, jamming, or countermeasures in hostile territories. GPS loss is a common phenomenon in the urban environment due to the interference caused by tall buildings and plenty of radiation. With these constraints, there is a need for alternate navigation (NAV) systems that are self-contained and passive. Image-guided NAV systems that rely on high-resolution cameras are ideal candidates under these constraints. This choice is further substantiated by recent advancements in computing resources, vision algorithms, and sensors that provide close to all-day weather capabilities. This work focuses on the aerial image matching problem for color images of the visible spectrum. Aerial images suffer from quality issues such as blurring, smearing, etc., caused by relative motion/angular disturbances of the acquisition platform and various atmospheric effects. These quality issues are mitigated to a certain extent by the high shutter speed offered by Charge-Coupled Device (CCD) cameras and the low integration time offered by Infra-Red (IR) cameras. However, these disturbances become more troublesome for high maneuvering aircraft and lightweight drones. A stabilized platform is required to contain these disturbances. Typical stabilization systems are built over a gimbal platform using gyros/inertial measurement unit (IMU), leading to drift over and above aircraft navigation sensors. The impact of the resultant drift is not linear. Control class gyros have higher drift than navigational class due to more accuracy requirements for the former than the latter. Image matching, therefore, becomes more challenging due to the variabilities in the acquisition platform (i.e., speed, attitude), disturbances (i.e., linear, angular), viewpoints (i.e., range, translation, rotation), multiple drifts, sensor characteristics, environmental conditions, time of acquisition, etc. It directs to the need for a self-reliant, robust, and efficient image matching technique.

Therefore, automatic image-based aerial NAV systems become very important, whose performance is dependent on robust and automatic real-time image matching. A few template images of the destination are required to guide the automatic aerial NAV systems. Ideally, these templates should have been acquired by another aircraft or aerial vehicle using an identical image sensor at the same atmospheric conditions/appearance and viewpoint. This constraint is challenging to meet in the case of remote or inaccessible locations and factors beyond our control. This difficulty is overcome by using widely available satellite images for template collection. However, there are a few challenges when working with satellite image templates. The critical challenges involved while matching a satellite image with live video (acquired using a belly-mounted camera on an aircraft) are the variations in view angle, atmospheric conditions, time-of-day, out-of-date images, and cross-platform (sensor) data, to name a few. Further, mismatches in the sensor wavelength and scene changes motivate us to address this as a cross-platform image matching problem.

To solve this problem, we present two contributions to this work. The first contribution is an enhanced cross-platform HD dataset with multiple image data galleries envisaging real scenarios and manually labeled ground truth. We present a methodology to augment an existing single platform aerial dataset with cross-platform imagery in addition to an efficient storage/retrieval mechanism. Our second contribution is a two-step robust aerial image matching framework consisting of coarse and fine stages. Coarse-matching builds over a modified pre-trained CNN with novel use of metadata. In contrast, fine-matching builds over state-of-the-art image matching methods to address the associated cross-platform matching challenges. We show that the proposed framework, though simple, can significantly improve the performance of image matching algorithms on our dataset, a derived dataset, and recently released cross-platform datasets. We demonstrate the efficacy of the matching framework using several standard metrics.

The rest of the paper is organized as follows: related work is discussed in Section II, and the enhanced dataset is presented in Section III. The proposed two-step matching framework is presented in Section IV. Results are deliberated in Section V, followed by concluding remarks in Section VI.

II. RELATED WORK

We briefly survey aerial image datasets and analyze state-of-art image matching approaches.

A. AERIAL IMAGE DATASETS

1) General-purpose Aerial Image Datasets

A few popular and publicly available aerial image datasets include HRSC2016 [1], DOTA [2], VHR-10 [3], SSDD [4] and so on. These datasets are built using satellite imagery [5] and address the object detection problem. INRIA [6], EuroSAT [7] and Drone datasets [8] are aerial semantic datasets with 2, 10 and 20+ semantic class labels respectively.

2) Geo-localization Aerial Image Datasets

Recently, there has been significant attention towards geo-localization of street view images. It implies the warping of aerial and satellite images over street view images. To improvise this process and accelerate the development of deep learning algorithms, several datasets are proposed in the literature [9]–[13]. Datasets covering the urban environment for such geo-localization tasks include the Zurich city [9], the Toronto city [10] and the work by Tian et al. [11]. We want to mention that these datasets are built over aerial or satellite imagery with slight cross-domain or cross-platform association. We address this shortcoming by proposing an enhanced version of our cross-platform path planning dataset [14]. Piasco et al. [15] explored the benefits of multiple types of heterogeneous data such as optical, geometric, and semantic. Our proposed dataset has multiple galleries, manual points correspondence, and semantic labels to represent each aspect [15] respectively.

3) Cross-Platform Aerial Image Datasets

Recently, Mughal et al. [16] and Zheng et al. [17] have released cross-platform Aerial Template Matching and University-1652 datasets, respectively, in the public domain. Mughal et al. [16] has created a multi-modal orthomosaic map by stitching aerial imagery acquired from low altitude MAV platform using a low-frame-rate camera in nadir view. The dataset comprises 2052 low-resolution aerial images (224×336) over 3 locations with multiple rounds. Authors [16] retrieved equivalent cross-platform satellite ortho map images from Bing. The dataset proposed by Mughal et al. [16] is similar in scope and aim to our proposed enhanced dataset of this work. However, University-1652 dataset [17] is intended to bridge the gap between ground-view and satellite-view by an intermediate aerial view for viewpoint-invariant feature learning. A synthetic drone-view camera simulates accurate flight, while images of 1652 buildings of 72 universities are extracted from the 3D Google Earth Engine (GEE [5]). The synthetic view camera retrieves 54 images of a building with three spiral rounds while the height descends gradually from 256 meters to 121.5 meters.

B. IMAGE MATCHING ALGORITHMS

With the advent of multiple types of aerial platforms and satellite imagery, as discussed above, visual place recognition (VPR) has become more challenging than image-retrieval due to the increased dimension of appearance and viewpoints and perceptual aliasing constraints. A comprehensive review of VPR and where is your place can be found in the literature [18], [19]. In the human visual system, place recognition happens by the firing of place-cells [18] or spatial view cells [19], which get triggered by sensory cues and self-motion. VPR [18] contains three major components, an image processing module to interpret incoming visual data (live query), a map to maintain representation (reference template), and a belief generation module (match algorithm). In this work, aerial DTV images constitute the live query followed by SAT target galleries which serve as the map reference template and the two-step matching framework, as the three major components in the same order [18].

To envisage path-planning objective, Courbon et al. [20] proposed a vision-based navigation strategy for vertical take-off and landing of UAVs using a single fish-eye camera and tested for indoor environments. It involves three steps, the first of which is building a memory of key image sequences followed by points detection using Harris corner detector [21]. Matching is carried out using a zero normalized correlation coefficient (ZNCC) around detected points. The authors [20] find the best fit actual image and follow the visual route in real-time. The similarity score is in proportion to the number of points detected by the corner detector. Martinez et al. [22] generated visual memory by simulating the navigation mission/path. A virtual model is generated using a robotic arm with a camera placed over a scene printed on a tarpaulin sheet. The matching of the onboard image and a desired virtual image is carried out using two quantitative metrics: the sum of squared differences (SSD) and mutual information (MI). Qualitative performance over the texture-less zone and outdated model is also discussed in this work. The authors [22] evaluated performance over different times of the day and ten-year-old scenes similar to proposed datasets dawn-dusk and historical galleries, respectively.

1) Classical Image Matching Methods

Several sparse and dense keypoint matching methodologies can be classified as traditional methods. There are keypoints detectors based on gradient, intensity, and blob. Harris [21] and Shi-Thomasi [23] corner detectors are gradient-based while FAST [24] is intensity-based. The efficiency of FAST and reliability of Harris detector formulate ORB detector [25]. The well-known SIFT [26] and SURF [27] algorithms are based on blob features exploiting second-order partial derivative (DoG). SIFT features [26] allow robust matching across different scene/object appearances, while discontinuity-preserving spatial mechanism allows matching of objects located at different parts of the scene. SURF is inspired partly by SIFT and is more robust under image transformations. SURF speeds up implementation by approximating Laplacian of Gaussian with a box filter and square-shaped filter for integral images. The SURF feature descriptor is built with the sum of the Haar wavelet response around the point of interest.

Classical feature-based descriptors are divided into local and global descriptors. SIFT, SURF, and FAST are examples of local descriptors, while color-histograms, HOG [28], GIST [29] are global descriptors. GIST uses Gabor filters at different orientations and frequencies to extract the 'gist' of the scene. Feature-based methods are relatively more efficient and can comfortably handle deformation up to a certain level. It requires a detection phase to indicate part of the image containing a detection tuned descriptor. Representative sparse keypoint detection algorithms include SIFT [26], SURF [27] and ORB [25]. Keypoints (or image features) are detected independently in the images, and corresponding keypoints are paired using the minimum distance between their features. Outliers are removed using the M-estimator SAmple and Consensus (MSAC) algorithm [30]. Geometrically transformed parameters are generated using the inliers or retained points. These geometric transformation matrices are used to warp one image over the other and calculate overlap parameters.

A good keypoint descriptor must be reliable, repeatable, and unique. Further, it must be invariant to rotation and variations in illumination. Traditional image matching relies on a local feature descriptor, global descriptor, or both. Global descriptors are more pose-dependent, while local descriptors are affected by lighting conditions. Matching local features [18] of one image with features of another is an inefficient approach. A bag of features technique is adopted for image retrieval inspired by the document search domain. Instead of using actual words/pictures, this method uses a bag of features/visual words to describe a document/image. A bag of words (BoW) captures each feature in a word form ignoring geometric or spatial structure, thereby representing it in reduced form. Bag-of-visual-word (BoVW) improves efficiency by quantizing SIFT or SURF descriptors into a vocabulary comprised of a finite number of visual words. Images with BoW description can be efficiently compared with well-established Hamming distance or histogram comparison methodologies. Further, an inverted index describing images improves storage efficiency. Additionally, topological maps with metric information – distance, direction, or both further improve retrieval performance.

Sivic et al. [31] proposed the Video Google concept in line with the standard text retrieval approach for object matching in videos. A set of viewpoint-invariant region descriptors represents an object in the video. It is analogous to text retrieval, where descriptors are pre-computed (vector quantized using K-means clustering) using inverted file systems. It reduces the impact of vector quantization and sealing. Philbin et al. [32] proposed a novel quantization method based on randomized trees using an efficient spatial verification stage to re-rank the results returned from the bag-of-words model derived using a 128-dimensional SIFT descriptor. The authors have manually labeled considering 11 landmarks images into good, ok (more than 25% present), junk (less than 25% present), and absent categories. Authors [32] have demonstrated improved performance across datasets.

Optical flow [33], SIFT Flow [34], DSP flow [35], Proposal flow [36] are a few well-known dense flow techniques in the literature. Optical flow [33] allows dense sampling in the time domain enabling target tracking, whereas dense sampling in space enables scene alignment. SIFT Flow [34] matches densely sampled, pixel-wise SIFT features between two images while preserving spatial discontinuities. DSP Flow [35] uses a deformable spatial pyramid (DSP) matching algorithm for computing dense pixel correspondences. Dense matching involves comparing the appearance between pixels and geometric smoothness between neighboring pixels. Unlike semantic flow approaches used in SIFT Flow and DSP flow, Proposal flow [36] exploits modern object proposals that exhibit high repeatability at multiple scales and can take advantage of both local and geometric consistency constraints among proposals. However, dense match algorithms are computationally very expensive. DeepMatching(DM) [37], inspired by deep CNN architectures, computes semi-dense correspondences between images. DM [37] is a robust technique based on a hierarchical, multi-layer correlation architecture. Further, it can handle non-rigid deformations and repetitive textures.

2) Deep Learning-based Image Matching Methods

Jiayi et al. [38] presented a detailed survey for image matching from handcrafted to in-depth features. It covers feature detectors/descriptors to matching methodologies with applications over Structure from Motion (SfM), Simultaneous Localisation and Mapping (SLAM), and image registration/fusion/retrieval methodologies. Deep learning-based detectors include LIFT [39] and Superpoint [40]. LIFT [39] is trained over SIFT with supervision from Structure from Motion (SfM), while Superpoint explores a fully convolutional model. The choice of the detector is task-specific. PCA-SIFT is a learning-based descriptor, whereas deep learning-based descriptor includes Siamese, triplet, and contrastive loss. SuperGlue [41] is a neural network that matches two sets of local features using joint correspondences and rejecting non-matchable points. Assignments are estimated by solving differential optimal transport problems using an attentional graph neural network. Authors [41] have introduced attention (self and cross) based flexible context aggregation mechanism. SuperGlue [41] is capable of working well with classical and learned features. Radiation-variation insensitive feature transform (RIFT) [42] is a feature matching algorithm that is robust to large Nonlinear Radiation Distortion (NRD). It uses phase congruency (PC) map for corner and edge points detection. RIFT builds feature descriptions using maximum index map (MIM). Finally, RIFT analyses the inherent influence of rotations over MIM for rotation invariance.

Xingyu et al. [43] presented a progressive filtering approach for feature matching by gridding the correspondence space and finding motion vectors. Outliers from the putative match set are discarded with density estimation of each sample and convolution of motion vectors. A coarse-fine strategy is adopted to refine motion vectors iteratively. Jiayi et al. [44] presented a two-class classification problem termed Learning for Mismatch Removal (LMR) with merely ten image pairs supervised. The authors [44] have established a methodology that is consistent with the consensus of the ratio of length and angle of motion vectors using an empirically chosen Gaussian penalty. Ren et al. [45] have proposed a Faster R-CNN approach for real-time object detection with Region Proposal Networks (RPN). Region proposals in the target scene are detected and localized with Faster R-CNN [45]. The Faster R-CNN architecture contains the Fast R-CNN as a detector network and the RPN as a region proposal algorithm. An RPN is a fully convolutional network that simultaneously predicts object bounds and objectness scores at each position. RPNs trained end-to-end to generate high-quality region proposals. A Siamese network built for unpaired and paired buildings using contrastive loss function. A graph constructed using local and global matches, while the final output is the mean of matched buildings. SimNet [46] is a neural network-based approach which exploits end-to-end trainable network to learn non-metric similarity functions for image retrieval. Features are extracted in a feed-forward manner by a pre-trained network. These features are fed to a visual similarity network for content-based image retrieval (CBIR).

Basu et al. [47] investigated deep learning methods for automatically extracting the locations of objects such as water resources, forests, and urban areas from given aerial images for applications in urban planning, forest management, climate modeling, etc. Weyand et al. [48] introduced the PlaNet model to achieve geo-localization using images. The authors [48] trained their model over a database of 126M images with Exif geo-locations mined from the internet. It performs well with landmark photos and delivers good performance with subtle geographical cues. The authors have experimented with sequence-to-sequence modeling using LSTM variants and reported accuracy improvement by 50%. Yang et al. [49] proposed matching of aerial images by extracting robust multi-scale feature descriptor using a CNN. Authors [49] upgraded VGG16 [50] network (e.g., the grid structure of 8×8), with feature extraction from the second, third, and fourth layers. Inliers are selected gradually to improve feature point registration.

A Neighbourhood consensus network (NCNet [51]) is a consensus network that learns local correspondences for match points without the need for global geometric constraints. It is an end-to-end trainable network built over features of ResNet50 [52]. Using an exhaustive pairwise cosine similarity match, it computes the correlation map (4D correlation tensor). These matches are filtered using soft mutual nearest neighbor filtering. It trains in a weakly supervised manner over the PF-Pascal dataset [53]. Recently, Mughal et al. [16] have proposed a trainable pipeline to localize aerial images in a pre-stored orthomosaic map. Further, the authors [16] have extended NCNet [51] a fully connected network (FCN)-based regressor for matching aerial images with satellite imagery. Authors [16] developed a framework mainly for intra-sensorial registration and tested for inter-sensorial (cross-platform) too. Cross-platform is a specific type of multi-modality. Jiang et al. [54] surveyed multi-modal image matching methods from area/feature-based to learning-based for various applications right from medical and remote sensing to vision. Authors [54] have explored 18 types of modalities, including cross-spectral (visible Vs. IR) and cross-temporal (outdated, time-of-the-day, season, etc.). Performance was evaluated using precision, recall, and f-score. Kong et al. [55] proposed a cross-domain image matching technique using deep feature maps.

Recently, Hausler et al. proposed Patch-NetVLAD [56] which is multi-scale fusion of locally-global descriptors for place recognition for street-view dataset with viewpoint and seasonal/time-of-the-day (e.g. dawn, dusk, night) appearance variation. The authors use the original NetVLAD [57] based descriptor to retrieve top-k matches and then compute patch descriptor using IntegralVLAD, followed by reordering of the initial match. The patch descriptors implicitly contain semantic information of the scene (e.g., building, window, tree, etc.) by covering a larger area. The authors proposed Rapid Spatial Scoring, which is an alternative to RANSAC [58] without the need for sampling.

Our proposed two-step matching strategy, which is a significant contribution to this work, is a combination of global features followed by local features. There are a few similar contributions in the literature [59]–[62]. Djenouri et al. [59], [60] proposed Decomposition Convolution Neural Network and vocabulary Forest (DCNN-vForest), where the first step does the extraction of regional and global CNN features. In the second step, these features are clustered using the K-means algorithm. The vocabulary tree vForest was formulated for each cluster's GPS-unavailable indoor industrial environment. Bai et al. [61] introduced a combination of Bag-of-word and deep neural network (BoWDNN). Yang et al. [62] proposed Hierarchical Deep Embedding (HDE) incorporating local features (SIFT), regional and global features (CNN) to construct a vocabulary tree of image database. Sunderhauf et al. [63] utilized ConvNet features as holistic image descriptors to analyze the robustness of different layers against appearance and viewpoint variance. The authors [63] concluded that mid-level features have robustness against appearance change. This work proposes a two-step matching framework where the first step uses global features and the second step exploits local features.

III. PROPOSED ENHANCEMENT OF CROSS-PLATFORM DATASETS

A. ENHANCEMENTS TO OUR PATH PLANNING DATASET

We first present the enhancements to our aerial cross-platform path planning dataset originally proposed in [14]. The reader is referred to [14] for a detailed description of the data collection experiment and the procedure for generating cross-platform aerial path planning data. For completeness, we briefly describe the experiment again, followed by an analysis of the proposed dataset. We acquired aerial imagery from a human-crewed aircraft at an altitude of about 4000'–5000' with an HD camera mounted at the aircraft's belly. This camera can acquire frames at a resolution of 1920×1080 at 60 frames per second (fps) and record in compressed form. We mounted navigation sensors to get instantaneous flight parameters. Instantaneous flight parameters include latitude, longitude, roll, pitch, heading, and altitude of aircraft. We use heading and altitude as extrinsic parameters during the coarse-match step, as will be described in section IV-A. This trajectory was transmitted to the ground via an RF link. Due to RF transmission, the data had a few noisy transients for various parameters. These transients are filtered with the expected profile of the aircraft sensor parameters. For cross-platform image generation, these filtered sensor parameters are used to generate the aircraft's trajectory that is encoded in a KML file. Historical data is retrieved from the Google Earth Engine (GEE) for the desired path. This satellite imagery will be referred as Satellite (SAT) in the rest of our discussion.

The enhancements to our path planning dataset [14] are in terms of improved alignment, resolution, and multiple historical galleries (offset, drift, and dawn/dusk galleries). These enhancements make the dataset well-suited for classification/VBL/VPR tasks in the cross-platform setting. We generated the aircraft trajectory by processing the metadata. We extracted corresponding images from GEE for the years 2009–2020. These historical images capture the effects of urbanization and atmospheric changes over the period. Images in the proposed dataset were acquired in the year 2016 by an aerial platform. To confirm this phenomenon with acquired aerial images from the year 2016, we applied standard 2D correlation and SSIM index [64] with SAT year-wise galleries (for grey and color images) as shown in Table 1. However, SSIM values were not consistent due to a lack of registration in our dataset [14]. The same is rectified up to a certain extent with fine manual alignment in this work. To further validate, we apply the keypoint matching algorithms (NCNet [51], RIFT [42], Patch-NetVLAD [56]). We report percentage of correct keypoints (PCK) in Table 2 for respective default thresholds (last column). It can be visualized from both tables 1 and 2 that correlation and PCK values degrade as we move away from the year of acquisition of DTV imagery (which is the year 2016). We describe these enhancements next.

1) SAT-Year-wise-Warped: Our existing dataset [14] has a qualitatively (i.e., visually) aligned set of images. In the existing dataset [14], we have carried out the fine-grained alignment for a few DTV query images over the SAT gallery. In this work, we have generated finely aligned frames for the entire gallery of 2500 images. We have manually marked points for a set of frames (DTV and SAT) at an interval of 10 frames. These points are tracked using a KLT-based points tracker over the frames and manually verified over each set. Then, the best five corresponding points are retained based on minimal re-projection error. The homography matrix is estimated and used to generate warped images with these correspondence points. The matching performance of warped images over the years is presented in Table 1. The average correlation for warped images improved significantly from around 0.15 [14] to 0.6 over the years. Similarly, the average SSIM score also improved over the years.
2) HD-DTV: This is the HD version of the three VGA DTV galleries in our path planning dataset [14] with 2500 frames per year.
3) HD-SAT-Yearwise: This is the HD version of SAT-Year-wise [14] gallery. It is similar to historical or outdated imagery of [22].
4) HD-Offset: Inertial navigation sensors (e.g., heading/yaw sensor) have bias issues. This gallery envisages positive and negative bias over the flight path. We have generated this gallery by adding constant shifts of 3.5 micro radians to the latitude. It has a constant offset with the SAT-Year-wise gallery of our path planning dataset [14] and the HD-SAT-Yearwise gallery of this dataset.
5) HD-Dawn-Dusk: Dawn and dusk galleries are representative of morning and evening time-of-the-day. Time-specific reference images are practically not feasible in real scenarios. In the existing dataset [14], DTV live query images are acquired at noontime, and it may happen that reference SAT images are available only for the morning or evening time. To envisage this scenario, we have created this HD-Dawn-Dusk gallery. This gallery is in the same spirit as learning representation from morning to the late afternoon by Lowry et al. [65].
6) HD-Drift: Gyros suffer from drift issues. Drift has a trade-off with accuracy/sensitivity, i.e., higher accuracy leads to increased drift. This gallery envisages a typical gyro's positive and negative drift over the flight path. We have generated this gallery by adding incremental shifts to the latitude over the path. We have maintained a typical drift rate of 0.5 deg per hour over the SAT-year-wise gallery of our path planning dataset [14] and the HD version presented in this work.
7) HD-1000-cross-platform: This is a set of 1000 HD DTV and SAT images. These images are derived from sequential video frames at regular intervals (every 30 frames), and equivalent SAT images are retrieved and visually aligned for each DTV query from GEE. We have carried out manual keypoints correspondence for each set of images, similar to the generation of warp gallery for existing dataset images. With this, we can generate fine aligned warped images. The 30 frames spacing enables the exploration of path planning for resource-constrained platforms.
8) HD-1000-segments: Semantic labels are marked manually for DTV and SAT images in 20+ classes. The labeling method is the same as described in our prior work [14].

B. ENHANCEMENTS TO THE OPEN-SOURCE UAV123 DATASET

In addition to enhancing our aerial path planning dataset, we also present improvements to the open-source UAV123 dataset [66]. We work with this dataset since aerial imagery (which we call DTV) is available from a drone platform for a known location. It has images of HD ready resolution (1280 × 720 pixels). This database was originally developed for tracking applications with over 110K images. It has nine major classes, including bike, boat, building, car, person, group of people, truck, UAV and wakerboard. Each major class (totaling 90) has sub-classes ranging from 3 to 26. The building class has five buildings or sub-folders. We selected three landmark images and retrieved equivalent images from GEE for these scenes with historical data for a few years. These three satellite images become the reference template for our matching framework. On further analysis of the dataset, we found that the classes are not independent. For example, the same scene appears in building and car categories. This is probably because images are acquired in the same region, and the dataset was originally meant for tracking applications where bounding boxes are needed. This mixed nature of objects makes it hard for image classification. Segregating the entire dataset manually for selected landmarks building is a tedious exercise that becomes even more difficult due to perceptual aliasing (similar buildings). We have carried out a semi-automatic procedure to circumvent these issues efficiently.

The steps in the semi-automatic procedure are described next. In these steps, the SAT images (GEE) form the query, and the DTV images [66] are the target galleries.

1) Manually select landmarks in the DTV target gallery [66] and retrieve the corresponding query SAT images from GEE [5].
2) Manually search best proxy match for selected landmarks in target DTV galleries. This step is manual due to the failure of automatic cross-platform matching carried out using algorithms such as RIFT [42] as shown in Fig. 1.

TABLE 1: Standard Match of DTV query with SAT-year-wise [14]/Augmented SAT-year-wise warped data. Correlation and SSIM values improve significantly with warped images and degrade as we move away from year of query DTV acquisition (as expected).

Metric	Year2009	Year2012	Year2016	Year2020	Remarks
Corr2 (grey)	0.100/0.57	0.139/**0.63**	**0.145/0.63**	0.083/0.62	Grey scale original images [14]/Warped images using corresponding points
Corr2 (color)	0.103/0.58	0.134/0.65	**0.149/0.66**	0.088/**0.66**	Color scale original images [14]/Warped images using corresponding points
SSIM (grey)	0.12/0.3	**0.25**/0.41	0.18/**0.42**	0.21/0.38	Grey scale original images [14]/Warped images using corresponding points
SSIM (color)	0.072/0.29	0.13/**0.4**	0.12/0.38	**0.15**/0.37	Color scale original images [14]/Warped images using corresponding points

TABLE 2: Performance of keypoint match algorithms over the years. First and second rows have threshold of 10 and 50 pixels respectively, while third row has both 10/50 pixels. PCK values degrade as move away from year of query DTV acquisition (as expected).

PCK	Year2009	Year2012	*Year2016*	Year2020	Remarks
RIFT [42]	11.8%	13%	**26**%	17%	Correct threshold 10 pixels
NCNet [51]	53%	69%	**93**%	79%	Correct threshold 50 pixels
Patch-NetVLAD [56]	11.36/34.52%	12.1/39.3%	**27.3/64.1**%	13.6/47.6%	Correct threshold 10/50 pixels

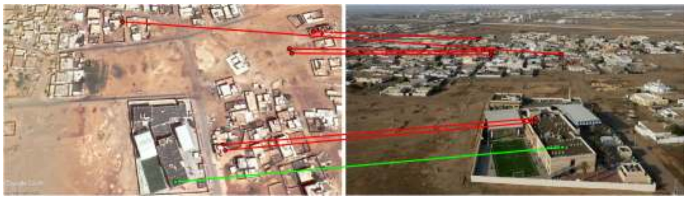


FIGURE 1: Cross-platform – SAT [5] and aerial [66] image matching [42]. An example of automatic match failure. Green and red lines imply true and false matches respectively. **Best viewed with zoom and color display**.

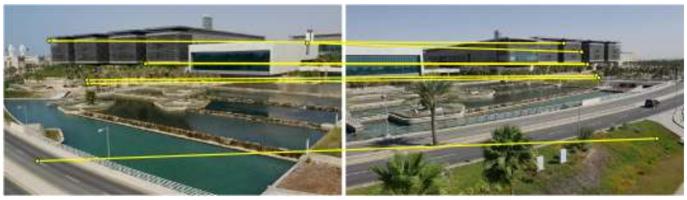


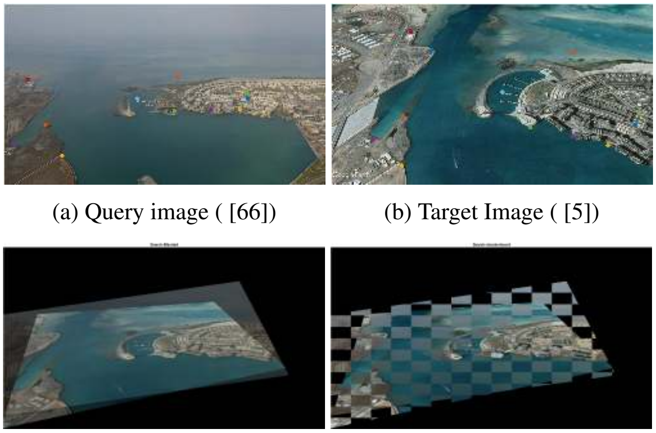
FIGURE 2: False scene match [42] due to perceptual aliasing from the UAV123 dataset [66]. Similar building both sides of center building. Yellow lines indicate locally correct match but globally incorrect. **Best viewed with zoom and color display**.

3) Once the proxy images are found, use an automatic matching method like RIFT [42] to find the best matches in the DTV target galleries. Automatic matching works here since images are taken from the same sensor.
4) Manually segregate the gallery into classes that best match the query SAT images. This step is manual to discard adversarial scenes (like similar buildings on both sides) as shown in Fig. 2.

We provide historical SAT imagery for landmarks. It is a test case for augmenting a standard dataset to have cross-platform capability.

C. DATASET ANALYSIS

A summary of the proposed enhancements and improvements over the baseline [14] are given in Tables 3 and 4 respectively. We now analyze the dataset to identify challenges involved in matching aerial imagery. We generate query-match profile curves for a few DTV query images manually. We first manually find the target-bin region for each query image, i.e., the set of frames in the SAT gallery containing the scene. A query and target image [66] are shown in Figures 3a and 3b respectively. We marked corresponding match points manually (colored dots). We use these points to find homography and generate the overlap image as shown in Fig. 3c. This overlap image can be visualized with the checkerboard in Fig. 3d for patchwise clarity.



(a) Query image ([66])

(b) Target Image ([5])

(c) Blended Queryimage

(d) Blended (checker board)

FIGURE 3: Manual marking of corresponding points (colored dots) and overlap representation. (a) Query image with colored manual points (b) Target image with corresponding manual points (c) Blended query image over target image (d) Blended image with checker representation. **Best viewed with zoom and color display**.

VPR [19] is the ability to recognize the overlap between two observations/images underlying match threshold con-

TABLE 3: A summary of the various galleries in the proposed enhancements to the datasets in [14] and [66].

Name	Resolution	# Images/Galleries	Description
SAT-Year-wise-Warped	640 × 480	2500 per year	Manually marked points for paired sequences [14] of DTVA-Reference and SAT-Year-wise images (2009-2020)
HD-DTV	1920 × 1080	2500 × 3	HD sequence of DTVA, DTVB and DTVC sequences of path planning dataset [14], 3 rounds
HD-SAT-Yearwise	1920 × 1080	2500 × 3 per year	HD sequence of SAT-Yearwise sequences (12 years, 2009–2020) of path planning dataset [14]
HD-Offset	1920 × 1080	2500 × 2 per year	Historical (12 years, 2009–2020) offset galleries with positive and negative bias
HD-Dawn-Dusk	1920 × 1080	2500 × 2 per year	Historical (12 years, 2009–2020) Dawn and dusk galleries at 6AM and 6PM local time respectively
HD-Drift	1920 × 1080	2500 × 2 per year	Historical (12 years, 2009–2020) drift galleries at rate of 0.5 deg per hour on either side of the ideal path
HD-1000-cross-platform	1920 × 1080	1000 × 2	Non-sequential (30 frames apart) High resolution DTV and SAT images with manually aligned match points
HD-1000-segments	1920 × 1080	1000	Semantic segmentation labels
UAV123-Cross-Platform-Classification	1280 × 720	3 galleries	Modified UAV123 dataset images [66] for classification with cross-domain images

straints. This overlap score for a query image against each SAT image in the target-bin region forms the query-match profile (also referred to as "normalized surface overlap" [67]). An example query-match profile curve with Gaussian fit is shown in Fig. 4 over the target-bin region. This is similar to representation of "ok" categorized images [32]. As can be seen, the best match score is 90% for a DTV query image over SAT gallery. It is normalized for better interpretation. An important takeaway from this query-match profile is that the overlap region varies smoothly over the SAT gallery frames and falls off almost symmetrically about the best fit. Query-match profile is normalized overlap over the target-bin region. Target-bin and query-match are the baselines for coarse and fine match performance evaluation, respectively.

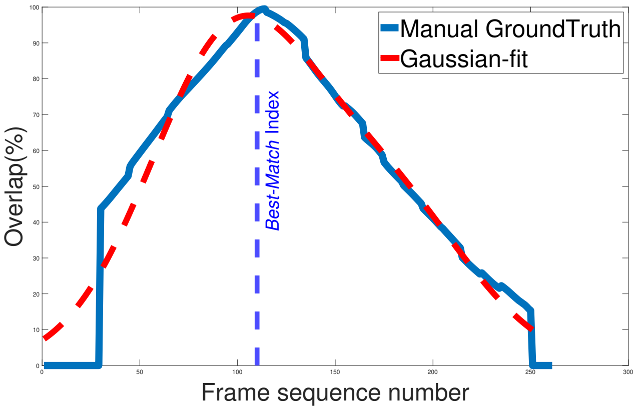


FIGURE 4: Query-match profile with Gaussian fit over target-bin. Normalized overlap curve built over manually marked corresponding points.

To further analyze the matching complexity of the proposed dataset, we applied various image matching methods over our SAT target gallery for one DTV query image. The performance of contemporary traditional and deep learning-based image matching methods for a DTV query image over the entire SAT gallery is shown in Fig. 5a and 5b respectively. Fig. 5a shows the performance of the Bag of visual words (BoVW) built over traditional descriptors. Fig. 5b represents the cosine distance between a DTV query and SAT target galleries. At the same time, the mean absolute error (MAE) for points deviation in the target-bin region for standard classical keypoints algorithms is shown in Fig. 5c. The difficulty with the existing traditional and deep learning methods is that they all have multiple maxima/minima over the search region. This necessitates the proposed coarse-fine matching technique, which exploits both methodologies appropriately. It provides the primary motivation for the proposed two-step image matching framework.

D. STORAGE/RETRIEVAL MECHANISM

The proposed dataset has multiple types of SAT galleries with historical information (12 years). Handling and storing HD images is very cumbersome. For ease [18], we propose to compress in video format along with embedding metadata in a structured manner. Image quality is retained reasonably with Spatio-temporal compression of sequential frames. We propose using time-stamp information for each frame indicating the type of gallery, year of acquisition, and frame number (in turn coordinates). Although, this information is available as metadata in standard video format (e.g., avi) and image format (e.g., GeoTIFF) but requires more space and an add-on utility. There are two components for embedding in the enhanced dataset: low and high frequencies. The low-frequency component includes the type of gallery and year number, while the high-frequency component involves frame indices. We embed low-frequency components in pixels and high-frequency components in bar patterns considering compression artifacts. We allocate four contiguous pixels per bar (per bit). High intensity (grey value of 235) and low intensity (grey value of 16) represent a logic ’1’ and logic ’0’, respectively. With this embedding, images extracted in png/jpg/bmp format will contain relevant information to automate frame processing for path, etc. The same embedding

TABLE 4: Dataset specific contribution

Parameter	**Baseline [14]**	**Proposed Upgradation**	**Remark**
Coarse Alignment	Yes	Yes	Qualitatively (visually aligned)
Fine Alignment	9 query images over target-bin region	2500 × 3 pairs	Quantitatively (manual corresponding points)
Sequential Images	Yes (2500)	Yes (2500 + 1000 images at an interval of 30 frames)	Inclusive of baseline
Aircraft Path Travel	8km	30km	Inclusive of baseline
Frame rate/Duration	60fps/125 Sec	2fps/500 Sec	Overlap reduced with low frame rate
Resolution	VGA (640 × 480)	HD (1920 × 1080)	Improved resolution
DTV segmentation	No	Yes	Manually labeled segments
Offset gallery	No	Yes (12 years)	Positive and negative offset bias galleries
Drift gallery	Yes (1 year)	Yes (12 years)	Left and right drifted galleries
Dawn/Dusk Galleries	No	Yes	Aids in improving generalization
Semantic Segments	SAT-BM gallery / 2500 × 1 (VGA)	SAT-Year-wise/2500 × 1 (VGA) + 1000 (HD)	Semi-automatic label transfer
Storage	Image	Video	
Retrieval/ordering	Image name with frame number	Encapsulated within video	Gallery name, year and index embedded.
Metadata embedding	No	Yes	
UAV123 [66] cross-platform enhancement	1) Forward SAT galleries 2) Backward SAT galleries 3) VGA resolution 4) Query image: UAV123 [66] 5) Target galleries: Augmented SAT	1) Landmarks Aggregation 2) Landmarks historical SAT images 3) HD-ready resolution (1280x720) 4) Query images: Augmented historical SAT images 5) Target galleries: UAV123 [66]	Generation of SAT target galleries to SAT historical query images
Applications	Path planning/Cross-platform matching	Path planning/Cross-platform matching/classification/Multiple galleries	Enhanced scope

can be extended further for other extrinsic parameters.

IV. PROPOSED TWO-STEP IMAGE MATCHING FRAMEWORK

The dataset generation process revealed several challenges in matching aerial imagery across different platforms or sensors. The most daunting of these include a lack of registration, mismatches in resolution, luminance, time of the day, perspective view variation, etc. Further, since the matching is to be done on low-resource aerial vehicles such as drones, this poses a further practical challenge. As we have seen in Figures 5a and 5b, the matching performance of traditional and deep methods, is not convincing. Based on discussions ([43], [44], [59]–[63]), global features describe the entire image and have more pose in-variance in contrast with local features describing patches (group of pixels). With this motivation and associated challenges, we present a two-step image matching framework, the first of which is a fast coarse-matching stage followed by a fine-matching stage. The former was built using a pre-trained CNN and the latter using off-the-shelf state-of-the-art matching methods.

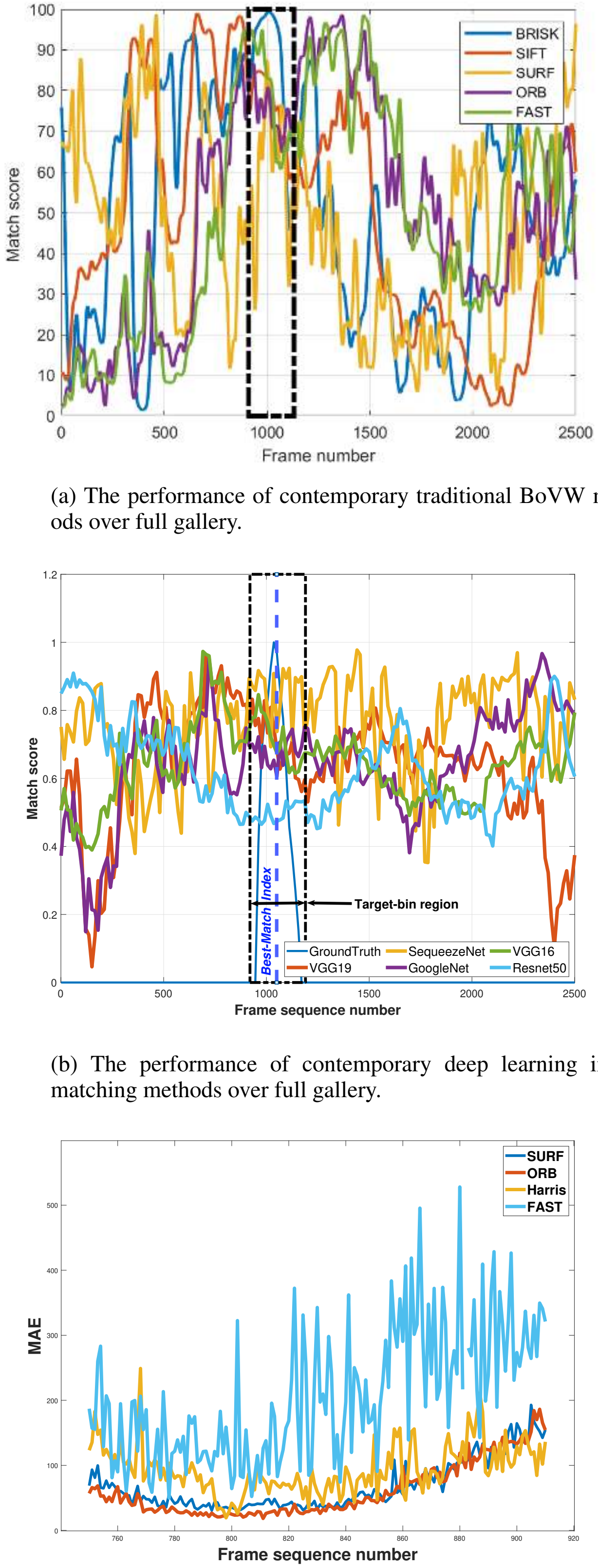
Our two-step matching framework is pictorially represented in Fig. 6. SAT images and features for the expected flight path and possible drifted paths are pre-loaded on the aerial vehicle as shown in the upper part of Fig. 6. SAT galleries are divided into bins (or temporally contiguous frames). The real-time query images from the onboard camera (called DTV images) are first fed to the coarse-matching stage. The coarse-matching stage indicates whether or not the input DTV query image is similar to the SAT target image. DTV frames whose classification indices (with the majority) correspond to the expected target-bin region or time instant are passed to the fine-matching stage. Standard keypoints matching algorithms are applied over the target-bin region to regress over the fine-match region. Outliers are discarded by exploiting *spectral*, *temporal*, and *flow* consistencies followed by cluster analysis. We describe this two-step matching framework in algorithm 1 and each of the stages in detail in the following sub-sections.

Algorithm 1 Proposed Two-stage Matching Algorithm

```
Input: Pre-trained model, GEE SAT image gallery of intended flight path, target_bin, and continuous stream of DTV data and extrinsic parameters.
while input DTV data stream available do
    Apply coarse-matching on incoming DTV frame and extrinsic parameters
    current_bin = output of coarse-matching stage
    if current_bin = target_bin then
        Apply fine-matching stage
        Output: Top-M matches and Confidence Score
    end if
end while
```

A. COARSE-MATCHING

The coarse-matching stage is designed to perform two functional tasks – reduce the overall computational complexity of matching and compare the actual flight path with the expected flight path. There are no constraints in terms of the need for very recent SAT imagery [5]. The coarse-matching



(a) The performance of contemporary traditional BoVW methods over full gallery.

(b) The performance of contemporary deep learning image matching methods over full gallery.

(c) Mean deviation error (MAE) in target-bin-region for classical image matching methods.

FIGURE 5: Performance of existing methods for a DTV query image over the SAT target gallery. The dashed box implies the target-bin region. The key takeaway from these plots is that none of the existing methods show satisfactory performance. **Best viewed with zoom and color display.**

step involves two phases – classifier fine-tuning carried out pre-flight (on the ground or off-line) followed by inference during the flight (onboard or real-time).

A deep learning classification model is fine-tuned pre-flight using the expected-flight path over SAT image galleries. These frames are first partitioned into N classes with M images in each class. Essentially, each class corresponds to frames from a temporally contiguous region in the flight path. We also designate the class(es) that contain the target or destination image(s) since this information is available pre-flight. We have experimented with different architectural modifications over pre-trained VGG16 [50] and ResNet50 [52] models. The pioneering work [63] concludes that mid-level CNN features exhibit robustness against appearance changes. In contrast, high-level features carry more semantics information and are more robust against changes in viewpoint. With this motivation of robustness against appearance changes and viewpoint, we empirically found that fine-tuning ResNet50 [52] with four appended dense layers (i.e., towards the end) and matching with features of a dense layer (fourth from last) performs reasonably well. The block diagram is shown in Fig. 7. Upper and lower branches of Fig. 7 represent pre-flight (ground-based or off-line) training activity and live (onboard or real-time) inference activity, respectively. The number of classes N is chosen so that each class has a sufficiently unique set of frames. In our experiments, we have chosen N to be 125 while the total number of images in the SAT gallery is 2500 (i.e., $M = 20$). The choice of N is dependent on factors including the flight speed and the frame rate. Once the fine-tuning is complete, we perform coarse-matching in the feature space of this fine-tuned model. The input DTV query image (onboard or real-time) is matched with a stored sequence of satellite images by extracting the features from the first appended dense layer. The mean square error (MSE) between the input DTV query image features and the satellite image features is then computed.

To further improve performance, we incorporate the real-time camera's extrinsic parameters available from the aircraft into our coarse-matching process as shown in Fig. 8. As discussed earlier, the DTV/airborne video is acquired along with extrinsic parameters (e.g., metadata) in real-time. The extrinsic parameters (altitude and rotation) complement the image level information. It is essential to consider this information to make the proposed approach more robust. We first relate the intrinsic parameters to the extrinsic parameters. To relate the camera field of view (FoV) to altitude, we generate an image for a typical altitude (with fixed known FoV). After that, we simulate incremental altitude in steps of 10m and generate images from GEE [5]. For typical altitude images and incremental altitude images, we detect SURF keypoints followed by standard outlier removal (e.g., RANSAC [58]) and generate the transformation matrix and the scale factor. This procedure generates an altitude to scale factor (zoom number) for the typical altitude. Similarly, rotation is the function of camera mounting and aircraft heading. We fol-

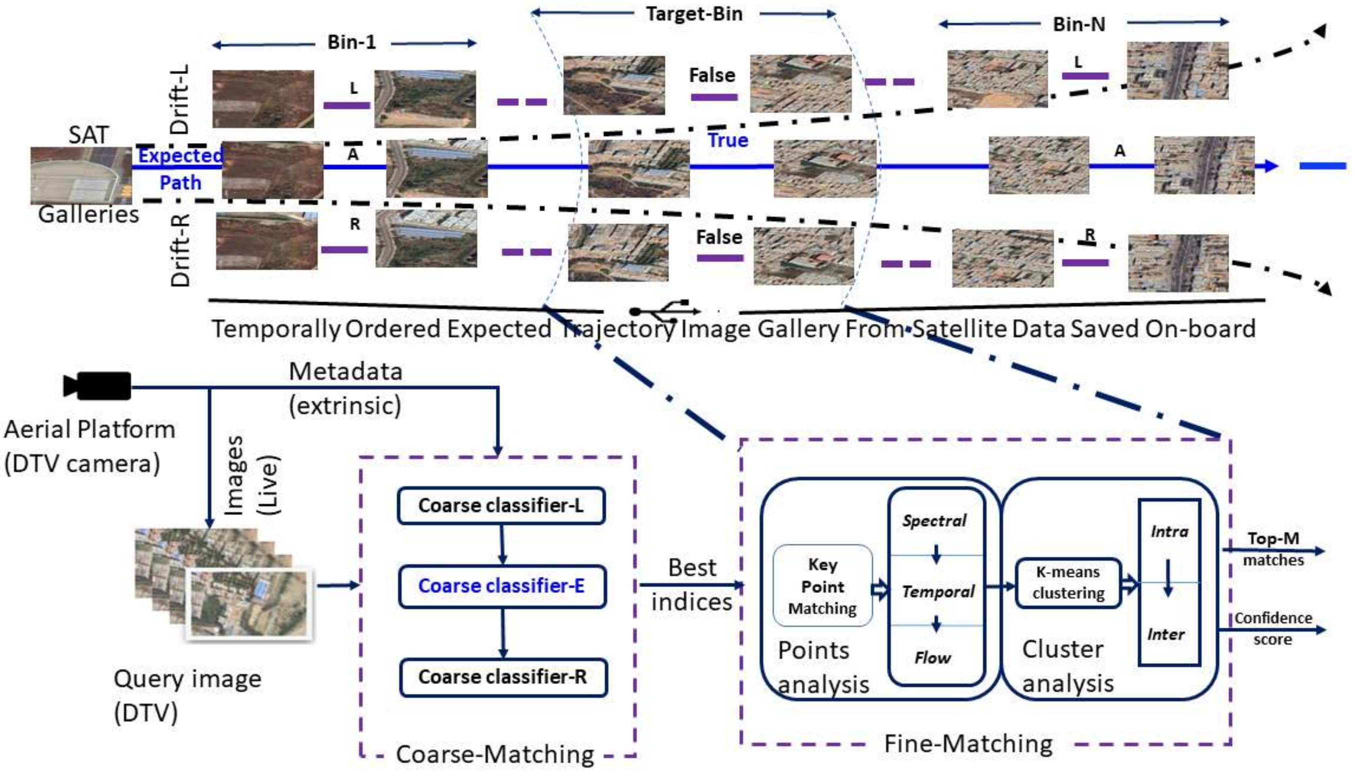


FIGURE 6: Proposed overall coarse-fine matching approach. Solid blue and dashed black lines imply expected and drifted trajectories, respectively. Coarse classifier-E fine-tuned over expected paths while coarse classifier-L/R are fine-tuned over left and right drifted paths, respectively. Fine-matching has points analysis followed by cluster analysis. **Best viewed with zoom and color display.**

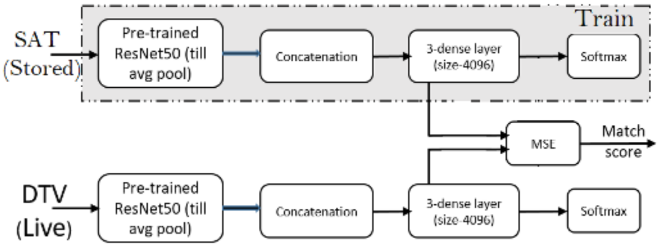


FIGURE 7: Proposed coarse-matching method with a modified ResNet50 [52] architecture (4 dense layers). Features are taken from first dense layer for testing.

lowed the same procedure as the altitude to scale factor for generating heading to rotation angle. Therefore, we have two extrinsic parameters per frame: altitude and rotation. For these extrinsic parameters to be effective, we found that they need to be projected to a higher dimension. This is found empirically using a dense network that progressively increases the dimension from 2 to 256. Specifically, this network has two hidden layers of sizes four and sixteen, followed by an output layer of size 256. We concatenate these extrinsic parameters along with the features from the pre-trained ResNet50 model [52] to fine-tune the model further. The block diagram of the proposed coarse-matching approach is shown in Fig. 8. The performance of the coarse-matching without and with extrinsic parameters is discussed in the next section.

B. FINE-MATCHING

The coarse-matching stage achieves two goals – one of checking the flight's expected trajectory and the other of identifying the target-bin region. Once the coarse-matching stage classifies the input frames as belonging to the target-bin region, all such frames are passed to the fine-matching stage. Additional checks are applied to such frames by comparing them with all SAT frames in the target-bin before declaring an overall match. The steps in the fine-matching stage include identifying corresponding match keypoints using a baseline method, performing consistency checks on the matched keypoints, clustering, and finally, confidence scoring. The fine-matching stage fundamentally leverages the temporal correlation in the gallery of images in the target-bin to prune out extraneous matching points. It helps the

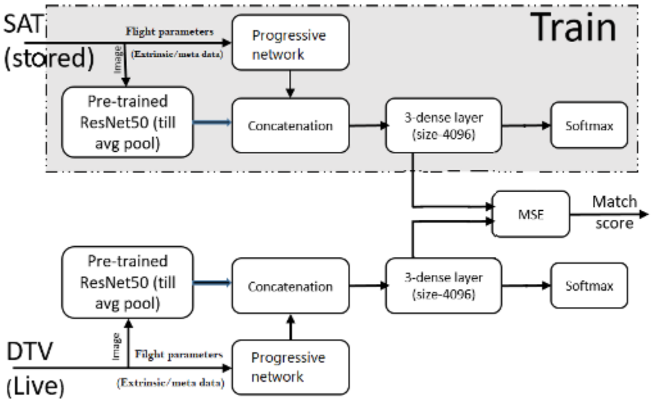


FIGURE 8: Proposed coarse-matching method (with extrinsic parameters). A combination of modified ResNet50 [52] architecture and extrinsic parameters are fine-tuned. Coarse-matching happens in the feature domain using MSE between the input DTV query image and target images of SAT gallery image features.

proposed method deliver match indices consistently and with higher confidence. We describe each of these steps next.

1) Key Points Matching

As the first step to finding matching keypoints, we apply the DeepMatching (DM) algorithm [37] to the DTV query image and the SAT images in the target-bin region of the SAT target gallery. Specifically, we find matching keypoints over five channels of the input image pair. These are the three color channels (Red, Green, and Blue), the average color channels, and luminance. DM [37], a semi-dense matching algorithm is applied to the corresponding pair of channels. DM [37] match for luminance channel are shown in the first column (1) of Fig. 9. We work with the semi-dense matching results to clearly illustrate the proposed pruning strategy. The semi-dense matching can easily be replaced by sparse matching methods such as SuperGlue [41] or RIFT [42] to find a sparser set of matching keypoints. We can even use a dense matching method like Deep Flow [68]. However, we have not tested the proposed strategy with dense matching methods given the resource-constrained environment where we expect our algorithm to be deployed. The semi-dense keypoints match correspondence becomes the input to the consistency check stage of our pruning strategy.

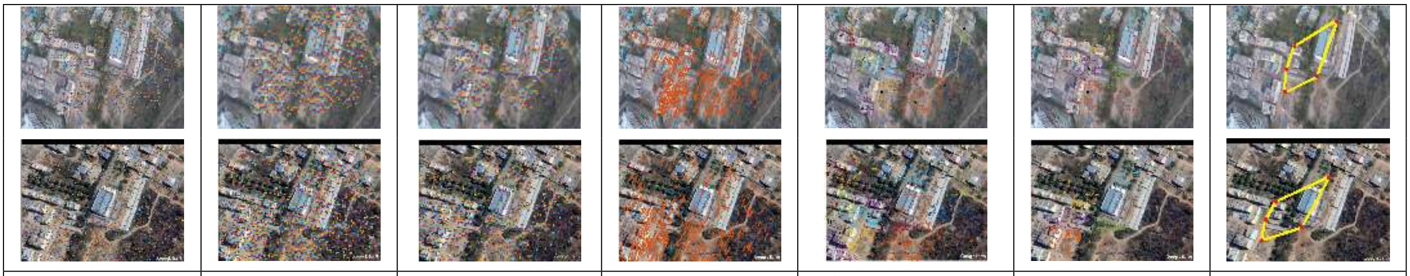
2) Points Analysis

We apply a series of consistency checks to prune further the keypoints identified by the semi-dense matching algorithm. The first of these is a check for *spectral* consistency. We hypothesize that for a keypoint to be consistent, it has to appear in a majority of the channels. In other words, a keypoint is declared to be consistent if it appears in at least three of the five channels over which DM [37] is applied. The resulting keypoints are used to create a mask which is then applied to the luminance channel keypoints. This set of masked luminance channel keypoints is used for further processing. Spectrally consistent points are shown in the second column (2) of Fig. 9.

We then apply a *temporal* check, which is based on the fact that matching keypoints must appear in the gallery for an expected number of frames depending on the speed of the aircraft, altitude, camera field of view (FoV), and the look angle of the acquisition platform. In our experiments, the vertical FoV of the camera is 25 deg and is looking down with a tilt of around 60 deg from the horizon due to mounting constraints (as stated earlier) with a target slant range of around 4000'. The aircraft speed is approximately 60 m per second, leading to the temporal displacement of around 1 m per frame at a camera frame rate of 60 fps. Due to resource constraint environment, we have down-sampled to VGA resolution [14] leading to a displacement of 2-3 pixels per frame. The same displacement is ascertained while finding a match between sequential frames. This displacement indicates scene or point appearance for 2.5 seconds on an average of 150 frames (at 60 fps). These 150 frames form the target-bin regions for the scene. In the target-bin region, a spectrally consistent point is expected to be appearing for at least 50% of the target-bin region temporally. All keypoints that do not satisfy this condition are pruned. Temporally consistent points are shown in the third column (3) of Fig. 9.

After the *spectral* and *temporal* checks, we apply a local motion check that is somewhat similar in essence [43], [44]. These works [43], [44] discard putative matches in the neighborhood by gridding correspondence space and consensus of length/angle using an empirical penalty, respectively. We claim that the optical flow at keypoints must be consistent with the average *flow* in their local neighborhood and propose a simple check for it. The optical flow of the entire set of satellite images in the target-bin is found a priori. The mean and variance of the optical flow magnitude (denoted μ_r, σ_r^2) and phase (denoted $\mu_\theta, \sigma_\theta^2$) around each pixel is then found over a voxel of size $8 \times 8 \times 8$. We find consistency in voxel in contrast with [44]. For every keypoint in the DTV image, we check if the flow vector at the corresponding match keypoint in the SAT image is consistent. By consistent, we mean that the magnitude and phase at the keypoint must lie within $\mu_r \pm 6\sigma_r$ and $\mu_\theta \pm 6\sigma_\theta$ respectively. All keypoints that fail this check are pruned. Flow consistent points are shown in the fourth column (4) of Fig. 9.

The corresponding improvements over the sequence of frames are shown in Fig. 10. Points analysis exploits *spectral*, *temporal*, and *flow* consistencies. The total number of consistent points and the accuracy over the frames at each stage is shown in Fig. 10 respectively. The number of match points reduces as we go away from the best match index due to the common region on either side. Base algorithm [37] has maximum points over the frames and reduces relatively with spectral/temporal/flow stages as shown in Fig. 10. The efficacy of remaining points, in target-bin region (i.e. frame no 135-255) is lowest over the frames for base algorithm [37]



(1) DM [37] → (2) *Spectral* → (3) *Temporal* → (4) *Flow* → (5) *Clustering* → (6) *Intra* → (7) *Inter*

FIGURE 9: Image matching outputs at various stages of the proposed algorithm. The algorithm proceeds sequentially from left to right. The top row corresponds to DTV images, and the bottom row corresponds to satellite images. Note the progressive improvement in the matching output along with the reduction in spurious matches. **Best viewed with zoom and color display.**

than spectral/temporal/flow consistent stages as shown in Fig. 10.

3) Cluster Analysis

The matching keypoints identified as consistent in the previous stage are clustered in the DTV query image using k-means Clustering. *Clustering* is motivated by the fact that matches will occur in a bunch of patches (specific features etc.) due to changes (appearance and viewpoints) over the period. We have empirically chosen the number of clusters to be 15 for our DTV query image in this stage. Consistent match keypoints in the SAT image are chosen to form another set of clusters (without applying k-means). Clustered points are shown in the fifth column (5) of Fig. 9.

Let the cluster index be denoted by c, number of keypoints in this cluster be N_c, and the keypoint locations in each DTV cluster be denoted by the matrix $\mathbf{K}^c = [\mathbf{k}_1^c, \ldots, \mathbf{k}_{N_c}^c]$ where $\mathbf{k}_i^c$ is the i^{th} 2D keypoint location in the c^{th} cluster. Further, let the centroid of this cluster be $\bar{\mathbf{k}}^c$. Similarly, let the corresponding SAT keypoint locations be denoted by the matrix $\mathbf{K'}^c = [\mathbf{k'}_1^c, \ldots, \mathbf{k'}_{N_c}^c]$ and the centroid of this cluster be $\bar{\mathbf{k}}'^c$. Also, let $\mathbf{D}^c = [\mathbf{d}_1^c, \ldots, \mathbf{d}_{N_c}^c] = [(\mathbf{k}_1^c - \bar{\mathbf{k}}^c), \ldots, (\mathbf{k}_{N_c}^c - \bar{\mathbf{k}}^c)]$ be the displacement matrix composed of the displacement vector of each keypoint in the DTV cluster relative to its centroid. The displacement matrix $\mathbf{D'}^c$ is defined in an identical fashion for the corresponding SAT image keypoints. The error matrix is defined to be $\mathbf{E}^c = \mathbf{D}^c - \mathbf{D'}^c$. This matrix captures the error between the displacement vectors corresponding to the DTV query and SAT target image keypoints. In the ideal case, this should be a matrix with all zero entries for every cluster c. However, this ideal case is very rare for oblique aerial and top-view outdated satellite imagery.

We propose the following strategy to identify the best matching keypoints clusters. The eigenvalues $\lambda_{\text{max}}^c, \lambda_{\text{min}}^c$ of the covariance matrix corresponding to the error vectors in $\mathbf{E}^c$ are found for each cluster c. We then pick those clusters c whose minimum eigenvalue λ_{min}^c is lower than a threshold τ that is found by applying the Otsu's algorithm over the set of minimum eigenvalues $\{\lambda_{\text{min}}^1, \ldots, \lambda_{\text{min}}^{15}\}$. This choice is guided by the fact that the minimum eigenvalue determines the ill-conditioning of a symmetric matrix and that the skewed displacement error matrix condition results in a highly ill-conditioned covariance matrix. Otsu's threshold is used since it helps in finding clusters that have minimum intra-class variance or, equivalently, maximum inter-class variance. *Intra*-cluster consistent points are shown in the sixth column (6) of Fig. 9. Once the best matching clusters are found, the corresponding keypoints are used to find the homography between the DTV query image and the SAT target image. The amount of overlap between the registered DTV image and the SAT image is the final matching score output by the proposed framework.

4) Confidence Score

We now present our approach to find the confidence score of the proposed matching framework. As discussed earlier, each query image has a target-bin region and a non-target-bin region in the SAT gallery. The confidence score is expected to be higher for the target-bin than the non-target-bin region. The keypoints passed from the previous step are used to find the confidence score. We cluster all passed keypoints into a few clusters in the DTV query image and corresponding group points in the SAT target image. The centroid of each cluster forms the vertex for the polygons in both images, as shown in the last column (7)of Fig. 9. We analyze across clusters (*inter* cluster analysis) by forming polygons in both the images. The polygons are unfolded [69] to calculate the turning radius [70]. The turning radius provides the mean squared error for the unfolded polygon but suffers from the issue of upper bound limit [70]. We experimented with a few ways of finding the score of turning radius and found that the weighted sum of the cosine of difference of turning angles, as shown in Eq. 1, outperformed the baseline [70] as shown in Table 5.

$$Score = \frac{1}{M} \sum_{i=1}^{M} W_i * \cos(\theta_i^c - \theta_i'^c), \qquad (1)$$

where $\theta_i^c, \theta_i'^c$ are the turning angles for the DTV and SAT cluster i; weight W_i: Points proportion for i^{th} cluster; M is the total number of clusters. Cosine of difference of subtended turning angles are weighed with points proportion of

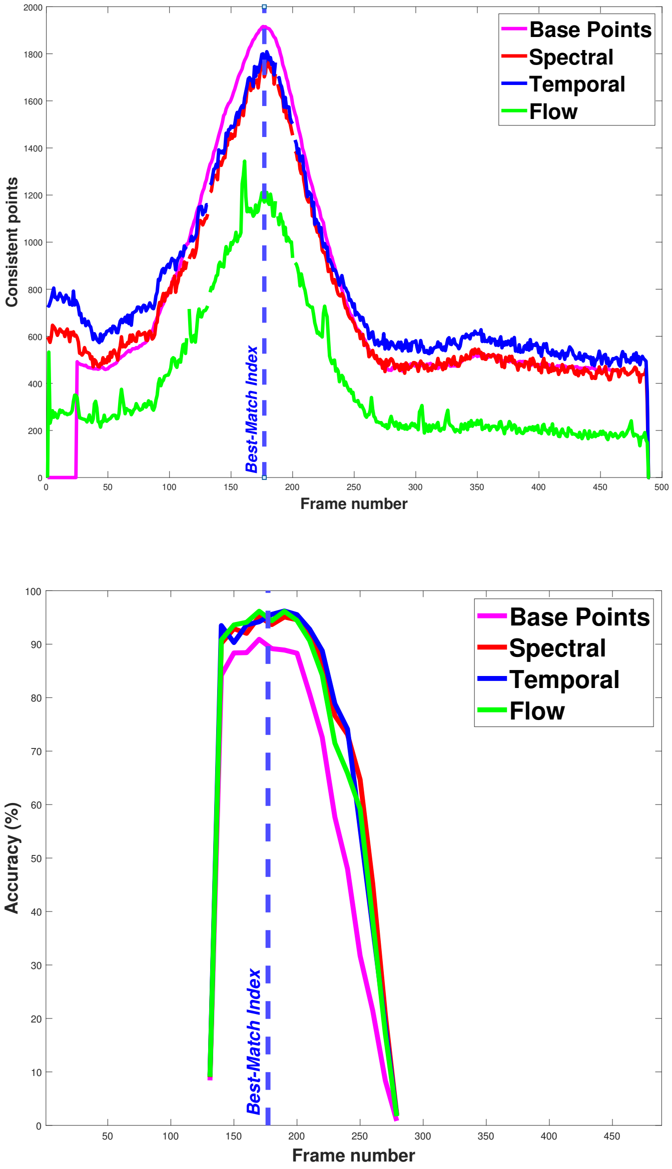


FIGURE 10: Improvements with proposed consistency analysis. (U) Number of consistent points over the frames. (D) Accuracy over the frames. We observe improvement in the accuracy over the baseline due to *spectral/temporal/flow* based filtering.

each cluster. This weighted sum is normalized for the number of clusters. From table 5, we can see an improvement in confidence score for the passed points over target-bin region than non-target-bin region. This significant improvement is due to false points being discarded by our procedure as described above.

V. RESULTS AND DISCUSSION

The proposed two-step aerial image matching framework is applied over the frames from the incoming DTV video sequence described previously. Given that our framework's first stage (coarse-match) performs standard image retrieval, it is evaluated using traditional image retrieval metrics (e.g., Top@N, mAP). The second stage, however, proposes methods to reduce outliers in state-of-the-art matching algorithms ([37], [41], [42], [56]). Therefore, we measure the performance of the second stage in terms of the improvement over baseline state-of-the-art matching methods (Naive/with RANSAC [58]). Further, the sequential aerial image matching application imposes additional matching requirements such as the percentage of overlap for the match and the Spatio-temporal accuracy of the match. Since there are no readily available metrics to measure this performance, we have adopted evaluation metrics [40] typically used in image matching and quality assessment. These evaluation metrics are briefly described as follows:

1) Top@N: Top N images retrieved for an input query image is widely used in the content-based image retrieval (CBIR) literature. It implies at least one correct match among the top N matching results from the SAT gallery. It is an increasing and monotonic function of N.
2) Mean average precision (mAP): mAP is the average precision over a bin. This is calculated over the full SAT target gallery.
3) F-score: Precision and recall represent the probability of correct detection for a class. High precision and recall are desired in matching algorithms. The F-score provides a combination of precision and recalls for unbalanced classes (i.e., target-bin region smaller than non-target region), and a higher F-score implies better performance. Searching for an image or short clip in a long video makes our classes imbalanced.
4) Percentage of correct keypoints (PCK): PCK is the standard way to designate the probability of correct keypoint [41], [71] on a set of matching images. It makes the underlying assumption of the threshold of correct match (e.g., the number of pixels/euclidean distance = $\alpha * \max(\text{width}, \text{height})$. α is a constant. A transformation matrix derived using the manual labeled keypoints is used as ground truth. PCK5 and PCK10 is described in Eq. 2 for $x = 5$ and 10 pixels respectively. This methodology is similar to Repeatability [40], [42].

$$PCKx = \frac{\left|\{\|Q_i^1 - \mathbf{H}S_i^2\| < x\}_{i=1}^{n_i}\right|}{N}, \tag{2}$$

TABLE 5: Performance of passed points using proposed methodology

Method	**Full gallery (2500 pairs)**	**Target-bin**	**non-target-bin**	**Remark**
Arkin et al. [70] (0 is best)	0.39	0.38	0.4	Target-bin region separation improved from 5% to 96%
Modified (1 is best)	0.12	0.998	0.046	

where $\mathbf{H}$ is the manually determined ground truth transformation between query Q and search S images; Q_i^1 and S_i^2 are the output paired coordinates of matching algorithm. N is a total number of paired points, and n_i is the pair index.

5) Mean Absolute Error (MAE): Li et al. [42] have used MAE for quantitative evaluation. It is the deviation concerning the homogeneous matrix. It is formulated in Eq. 3 with the transformation matrix built over manually marked correspondence points.

$$MAE_j = \frac{\sum_{i=1}^{N} \left|Q_i^1 - H_j S_i^2\right|}{N}, \tag{3}$$

where, H_j is manual transformation matrix for query image with j^{th} index image in target SAT gallery. Q_i^1 and S_i^2 are the output paired coordinates of matching algorithm. N is the total number of paired points and i is the pair index. This methodology is similar to Localisation error [40].

6) Ratio-metrics (RM): Ratio-metrics [42] is calculated over the match pairs temporally. It describes the ratio of the number of image pairs with correct match pairs above 50% (PCK threshold of 10) over the entire gallery. It is described in Eq. 4. It gives a quick idea about the performance of a query image over the target-bin region (the majority of correct matches in image pairs).

$$RM, r_{PCKx>50\%} = \frac{\left|\{PCK_x > 50\%\}_1^{N_G}\right|}{N_G}, \tag{4}$$

where N_G is no of images in target-bin region of a gallery and x is 10 pixels.

7) Overlap: The amount of overlap [72] between query and target image is another way of evaluation. This overlap is derived from the homography of matched points. This methodology is similar to homography estimation [40]. Query-specific query-match profile curve is the ground truth. The overlap is quantified using the following metrics.
   a) Positional accuracy (PA): PA represents how close the match index (i.e., time instant) is to the peak of the query-match profile curve (*Best-Match* index) as shown in Fig. 4. This curve is the basis for finding the score for the given index. Predicted indices close to the peak of the query-match profile will have better positional accuracy.
   b) Pearson's Linear Correlation Coefficient (PLCC): PLCC indicates a linear correlation between two sets of values. It is also termed the normalized correlation coefficient. It varies between +1 and -1. Correlation between predicted query-match profile in the SAT gallery is compared with manual query-match profile curve (Ground truth as shown in Fig. 4).
   c) Spearman Rank Ordered Correlation Coefficient (SROCC): SROCC is the non-parametric measure of rank correlation. It measures the temporal relation between predicted and ground truth data. It finds a correlation in the rank order given by the matching algorithm against the query-match profile. Higher SROCC is indicative of better performance.

A. RESULTS

We present the results of our proposed matching framework in a stage-wise manner with coarse-matching results followed by those for fine-matching. We have divided the SAT galleries into two parts, target-bin region (query image available - OK [32]) and non-target-bin region (no match feasible with the query - Absent [32]). Coarse-matching performance is evaluated over the entire SAT gallery (i.e., 2500 frames) consisting of both regions. In contrast, fine-matching performance is evaluated over a target-bin region as indicated by the dashed box in Fig. 5a and 5b.

1) Performance over entire gallery

We evaluate the performance of the proposed coarse classifier with state-of-the-art methods. These contemporary methods have demonstrated excellent Performance on several standard image matching datasets. We evaluate Performance using standard matching methodologies as described earlier. A match is considered valid when the *Match* index from the matching algorithm lies in the target-bin region of the query image.

Specifically, we carried out a standard image search experiment (Fig. 5) for a few query images (DTV) in several example target galleries (SAT). We have considered a few state-of-the-art image matching algorithms for comparison. These include both conventional and recent deep methods. For conventional methods bag of visual words with descriptors, [25]–[27] are used for comparison. Pretrained models (VGG16 [50] and ResNet50 [52]) are used in the inference mode to find the difference between the query and target images. For SimNet [46] and CNN-registration [49], we used the available online implementations.

Top@1, Top@5, Top@10, Top@20 and mean Average Precision (mAP) for the proposed and contemporary methods are summarized in Table 6. We have used 9 DTV query images and tested with SAT year-wise target galleries [14].

TABLE 6: Performance of Coarse-matching over SAT-year-wise galleries (Higher is better)

No.	**Method**	**Top@1**	**Top@5**	**Top@10**	**Top@20**	**mAP**
1.	BoVW (SIFT [26])	22.2	27.8	27.8	33.3	19.95
2.	BoVW (SURF [27])	19.4	47.2	55	69	21
3.	BoVW (ORB [25])	19.4	38.8	41.6	44.4	15.28
4.	BoG Spatial [31], [32]	22.2 [14]	38.8	41.6	50	**23.9**
5.	CNN-registration [49]	7 [14]	50	50	**71**	15
6.	SimNet [46]	27.5 [14]	38.4	49.5	57.75	14.4
7.	Pretrained VGG16 [50]	27.7 [14]	36.1	41.6	42	23.1
8.	Pretrained ResNet50 [52]	16.6 [14]	19.4	22.2	30	18
9.	*Proposed fine-tuned Classifier*	41.4	50	55.5	55.5	22.3
10.	*Proposed fine-tuned Classifier with Extrinsic Parameters*	**44.4**	**52.7**	**58.3**	61.1	23.4

The mAP is calculated for Recall 1. Performance of BoVW vocabulary built over local features ([25]–[27]) and global features ([31], [32]). BoG spatial ([31], [32]) performs a bit better in terms of mAP but poor for all Top@N retrievals. Baseline, VGG16 [50] pretrained network performs better than ResNet50 [52] for all parameters. CNN-registration [49] and SimNet [46] work reasonably well for Top@10 and Top@20 but have low mAP. The proposed classifier with extrinsic parameters demonstrates consistent improvement concerning Top@N retrievals and mAP. As stated in the coarse-matching description (section IV-A), we empirically found that fine-tuning ResNet50 architecture [52] performed well. The proposed fine-tuned architecture without/with extrinsic parameters performs considerably better than the baseline model. Recall and F-score curves for Top@N retrieved indices considering baseline, fine-tuned architecture without and with extrinsic parameters are shown in Fig. 11a and 11c respectively. Similarly, Precision-Recall showed in Fig. 11b. The fine-tuned architecture is better than the baseline, and it further improves with extrinsic parameters. Extrinsic parameters are readily available (metadata) for any aerial journey, and the role of extrinsic parameters in improving overall performance is clear from these results.

2) Performance over target-bin region

With clues from coarse-matching stage, fine-matching is carried out over expected target-bin region as shown in Fig. 6. The former outputs a few probable frames (i.e. indices) which are validated by the latter one. As stated, we built over standard matching algorithms (DeepMatch [37], SuperGlue [41], RIFT [42], Patch-NetVLAD [56]). We compare against baseline ([37], [41], [42], [56]) without/with standard outlier [58].

To evaluate the performance of the fine-matching stage, we test it over the target-bin region and apply standard metrics like PCK, MAE, ratio-metric (RM), etc. In the entire target-bin region, the query image is assumed to be available at least partially (i.e., OK [32]). PCK5 and PCK10 imply designated deviation within 5 and 10 pixels, respectively, from the ground truth correspondence. Fine-matching performance over target-bin region is tabulated in table 7. We notice a clear improvement in PCK5 and PCK10 over the baseline and with standard outlier (RANSAC [58]) implementation. PCK10 is expected to be better than PCK5. The same is validated from columns 3 and 4 of table 7 for base methods ([37], [41], [42], [56]), along with RANSAC [58] and proposed outlier removal methodology. The mean deviation error for matched points is shown in Fig. 12a for DTV query image in SAT target gallery [14]. This deviation increases as we go away from *Best-Match* index in the target-bin region. For SuperGlue [41], baseline deviation error increases further with RANSAC outlier as depicted with PCK values reduction in table 7. For the proposed outlier removal, the mean deviation curve is reasonably low and flat for the entire target-bin region, as shown in Fig. 12a. Matched points precision for varying euclidean distance threshold is shown in Fig. 12b. Precision is a monotonically increasing function with an increase in the threshold for all methods as expected. Improvement is clear for proposed methodology over multiple baselines([37], [41], [42], [56]).

Matchpoints (pairs) are used to generate overlap using homography. A higher overlap is indicative of a better match. To search a DTV query image in a target SAT gallery, the ideal output should have an inverted ‘V’ shape, i.e., a clear peak (corresponding to the highest overlap) at the correct matching location. It should quickly taper off as we move away from this ideal location. The proposed method is compared with contemporary methods in terms of overlap percentage over the target-bin region as shown in Fig. 12c. From this figure, we see that baselines ([37], [41], [42], [56]) overlap and have multiple peaks/valleys towards the endpoint (right side of curves), which is subdued by the proposed outlier removal approach. Additionally, the inverted ’V’ shape output (query-match profile) with a peak close to the *Best-Match* index is also very evident for the proposed method. We use PLCC and SROCC to quantify this overlap performance relative to the manual ground truth (query-match profile as shown in Fig. 4) quantitatively. Despite the poor performance of SuperGlue [41] with RANSAC [58] in terms of PCK, it performs well in terms of PLCC and SROCC. PA relates predicted best frame (i.e., time instant) deviation from *Best-Match* index as shown in Fig. 5b,12a,12c. PA for DM [37] with RANSAC [58] is reasonably high relative to the other methods.

The points passed by the proposed approach for a query image in the target-bin region are shown in Figures 13a, 13b, 13c, 13d over standard methods [37], [41], [42], [56] respectively. True and false matches are represented with

TABLE 7: Fine-matching performance over the target-bin region. Bold numbers imply the best performance for the given baseline. Except for the last column, higher is better.

No.	**Method**	**PCK5 (%)**	**PCK10 (%)**	**PLCC**	**SROCC**	**PA**	**RM**	**MAE**
1.	DeepMatching [37] (Baseline)	18.8	35.9	0.23	0.26	68.1	18.42	79.3
2.	DeepMatching [37] + RANSAC [58]	10.4	21.4	0.43	0.43	**80.4**	14.10	159.8
3.	*DeepMatching [37] + Proposed*	**25.3**	**42.4**	**0.76**	**0.76**	73.07	**28.21**	**76.9**
4.	SuperGlue [41](Baseline)	2.17	8.03	0.59	**0.66**	60.9	3.39	159.7
5.	SuperGlue [41] + RANSAC [58]	1.03	4.46	0.62	0.6	73.2	0.99	217.1
6.	*SuperGlue [41] + Proposed*	**7.55**	**23.8**	**0.63**	0.6	**82.8**	**10.02**	**132.6**
7.	RIFT [42] (Baseline)	1.1	3.52	0.55	0.58	77.6	1.24	146.8
8.	RIFT [42] + RANSAC [58]	1.3	4.17	0.57	0.57	77.8	1.35	164.2
9.	*RIFT [42] + Proposed*	**18.35**	**35.79**	**0.72**	**0.62**	**83.2**	**36.29**	**55.13**
10.	Patch-NetVLAD [56] (Baseline)	0.81	3.01	0.32	0.32	60.07	0.92	214.4
11.	Patch-NetVLAD [56] + RANSAC [58]	0.62	2.19	0.52	0.52	67.4	0.73	276.08
12.	*Patch-NetVLAD [56] + Proposed*	**10.3**	**31.9**	**0.56**	**0.58**	**77.3**	**59.8**	**10.9**

TABLE 8: Matching performance over the UAV123 dataset [66]. Bold numbers imply the best performance for the given baseline. Except for the last column, higher is better.

No.	**Method**	**PCK5(%)**	**PCK10(%)**	**PLCC**	**SROCC**	**PA(%)**	**RM**	**MAE**
1.	DeepMatching [37] (Baseline)	22.9	49.2	0.2	0.36	53	22.5	57
2.	DeepMatching [37] + RANSAC [58]	8.9	22.3	**0.56**	**0.59**	**70**	9.9	37
3.	*DeepMatching [37] + Proposed*	**32.9**	**63.1**	0.27	0.15	64	**23.8**	**26**
4.	SuperGlue [41] (Baseline)	3.5	11.2	0.51	0.61	**82**	0.38	99
5.	SuperGlue [41] + RANSAC [58]	**10.9**	12.3	0.71	0.69	60	0.97	193
6.	*SuperGlue [41] + Proposed*	10.8	**15.8**	**0.84**	**0.78**	81	**10.5**	**34**
7.	RIFT [42] (Baseline)	2.0	4.8	**0.48**	0.36	64	0.2	201
8.	RIFT [42] + RANSAC [58]	2.1	5.3	0.42	**0.55**	64	0.21	200
9.	*RIFT [42] + Proposed*	**2.69**	**6.8**	0.25	0.15	**81**	**3.4**	**45**

TABLE 9: Computational complexity break-up of the coarse and fine matching stages of the proposed algorithm

Coarse-match (Inference)	**Fine-match**			**Remark**
	Standard matching method ([37] / [41] / [42])	Points analysis (*Spectral / Temporal / Flow*)	Cluster analysis	
80 mSec	9 s / 3 s / 7 s	0.2 s / 0.7 s / 0.8 s	78 mSec	CPU

green and red colors, respectively, to visualize the efficacy of the proposed approach for a query image in the target-bin region. These figures show that the remaining (leftover) false matches (PCK5 constraints) are very few. False matches (red color) are further reduced with PCK10 constraints as expected. The performance of the proposed and contemporary methods can be visualized qualitatively in Fig. 14 and Fig. 15 for Top@1 retrieved image of coarse-match and fine-match respectively over the years. Coarse-matching retrieved images are distributed all over the gallery, and the same is depicted from the Fig. 14. These figures show that the retrieved images for a query image over the years are relatively consistent with the proposed approach. Fig. 15 represents a fine-match performance over the target-bin region, and therefore, the retrieved images are pretty similar for all methods. It is clear again that retrieval performance improved for the proposed outlier rejection over the years. This improvement is relatively hard to visualize since the retrieval is within the target-bin region. From these figures and scores, it is clear that the proposed coarse-fine matching method delivers improved performance.

Further, we evaluate the proposed approach over the modified [14] UAV123 dataset [66] and report the performance in table 8. We want to reiterate that the query image is SAT, and the target image is DTV [66] here. Generated forward and backward galleries [14] have only target-bin region; hence only fine-matching performance is evaluated. Table 9 quantifies the typical computational complexity of various associated modules. The computational efficiency of the proposed approach can be seen from these numbers for coarse and fine match components.

TABLE 10: Matching performance over the Aerial Template Matching dataset [16]. Except for the last column, higher is better.

Method	**PCK5**	**PCK10**	**RM**	**MAE**
RIFT [42] (Baseline)	6.17	15.43	7.22	38.63
RIFT [42] + RANSAC [58]	4	6.17	7.22	40.4
RIFT [42] + Proposed	**14.81**	**41.9**	**17.39**	**21.08**

TABLE 11: Matching performance over the University dataset [17]. Except for the last column, higher is better.

Method	**PCK5**	**PCK10**	**RM**	**MAE**
RIFT [42] (Baseline)	34.08	63.1	23	12.4
RIFT [42] + RANSAC [58]	29.06	56.4	19	13.39
RIFT [42] + Proposed	**36.1**	**64.8**	**26**	**11.7**

For further validation, we have carried out initial experiments with a recent Aerial Template Matching [16] and

TABLE 12: Cross-platform dataset comparison

Dataset	Images	Resolution	Platform	Video	Type	Application	Altitude	Variation	Coverage
University-1652 [17]	1,46,593	512 × 512	GEE, GEE-3D	No	Synthetic	Classification	121.5-256 m	54 views	1652 buildings
Aerial Template Matching [16]	2052	336 × 224	Bing, Aerial	Yes	Real	Recognition	2000 ft	3 areas	2 km^2
Path Planning [14]	25028	640 × 480	GEE, Aerial	Yes	Real	Recognition	5000 ft	Historical galleries	8 km
Cross-platform HD	32000	1920 × 1080	GEE, Aerial	Yes	Real	Recognition, Classification	5000 ft	Multiple galleries	30 km

synthetic dataset [17]. As discussed earlier, dataset [16] has low resolution and low frame rate aerial images (DTV). The target-bin region contains 4 or 5 frames. Originally, the SAT image [16] was a map from Bing, whereas we retrieved SAT image for the same area from GEE and named it a query SAT image. The aerial images from [16] are named as target images/galleries. As discussed earlier, we marked corresponding points manually for the target-bin region (4-5 frames) to generate a query-match profile. Due to the small target-bin region, we apply the proposed cluster-analysis of the fine-match step using RIFT [42]. The University-1652 dataset has many buildings with 54 views for each in a synthesized manner. We manually marked corresponding points for a few buildings. As above, we apply cluster analysis of proposed fine-matching using RIFT [42]. Performance is evaluated against baseline [42] without and with standard outlier [58], for aerial template [16] and university [17] datasets in table 10 and 11 respectively. Improvement is clear from these tables regarding PCK, MAE, and RM.

B. DISCUSSION

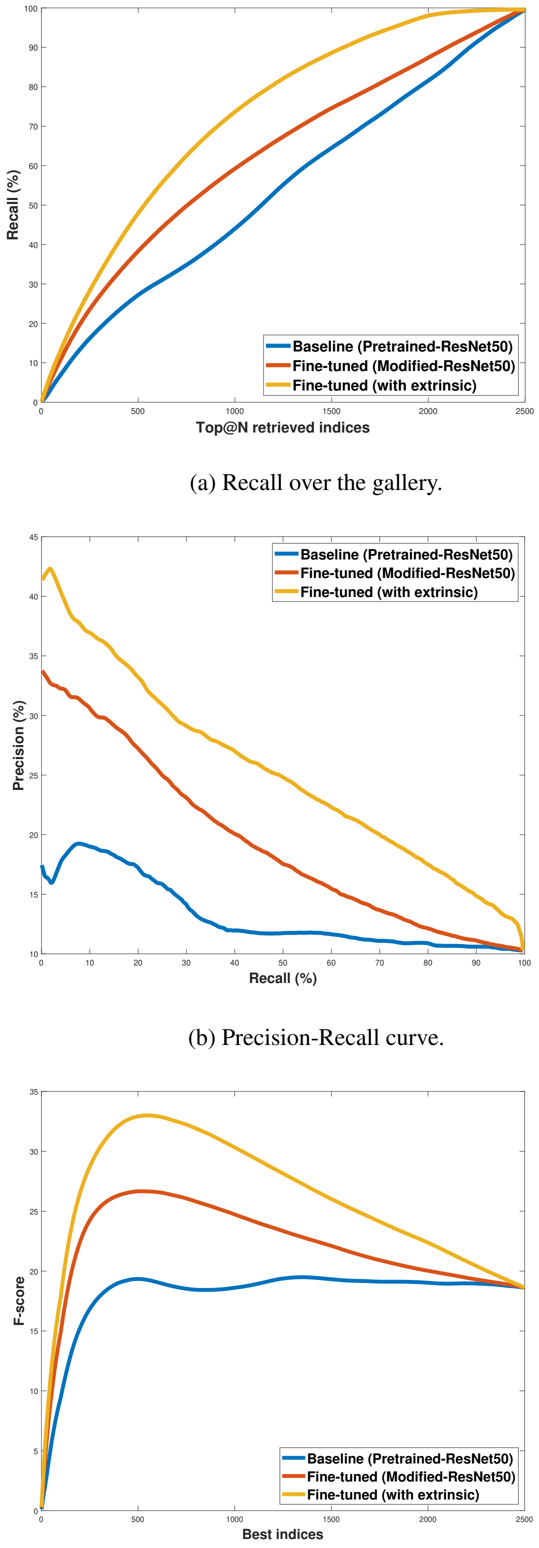
This section briefly discusses our dataset enhancement and the two-step matching framework. We enhanced the dataset with more realistic scenarios (HD images, manual labels, drift, offset, and dawn/dusk galleries) as summarized in Table 3. Enhancements over our earlier dataset [14] is presented in Table 4 to clearly highlight the contributions of this work. Offset and drift galleries are generated by adding latitude since aircraft had to travel longitudinally (runway "East-West" constraints). We have compared proposed enhancement with publicly available cross-platform aerial datasets ([16], [17]) in Table 12.

The corresponding sets of images are finely aligned with manual point marking, and its efficacy is shown in Table 1. Due to the aligned nature of SAT-Year-wise-Warped and HD-Dawn-Dusk galleries, the same can be used to train the network [65] to generate unseen galleries. From Figs. 14 and 15, we see that the proposed dataset covers urbanization over the period, atmospheric distortions (for e.g., small clouds) and so on. As an improvement, the dataset can be further enhanced with topological-metric [18] describing objects and their interrelations. Seasonal and night galleries may be further appended. For storage and retrieval, semantic compression with metadata embeds and inverted matrix [31] shall be explored.

VPR research is challenging due to the lack of a standard definition of 'place' and various datasets with varying metrics. Typically, the VPR problem revolves around landscape/landmark/place while missing the aerial image aspect. The proposed two-step matching framework presents a coarse-fine approach for aerial image matching. The CNN-based coarse-matching stage is fast, efficient, and accurate and is ideally suited for the low resources feasible onboard platforms. The tunable parameters include the number of classes N and the number of bins M (and, therefore, the number of images K in a bin). These parameters can be chosen based on the speed and altitude of the aircraft. Extrinsic parameters are a step toward multi-sensor data fusion for real-time applications. We have built over a stable, popular, and well-accepted network [52] as a baseline. It can be further improved by using the latest models. Optimal arbitration logic to use indices of multiple classifiers shall be explored. The fine-matching stage builds on state-of-the-art image matching methods ([37], [41], [42], [56]). We leverage over the points and cluster analysis to improve matching performance. We exploited 3D information for outlier removal in contrast with 2D based [43], [44]. We have demonstrated the efficacy of both stages using several evaluation metrics. Instead of sticking to image features [16], we have leveraged *spectral*, *temporal*, and *flow* features. The state-of-art datasets [16], [17] have been tested as an initial step using one matching algorithm [42] as a baseline. We plan to extend this for the entire framework with multiple matching algorithms. Additionally, 3D scene modeling along with metadata shall be explored. Matching a satellite image with a thermal image is another line of research (due to drastic texture variation) for practical applications (day-night applications). Aircraft with gimballed cameras give flexibility to focused surveillance, but nonlinear combination results in a wide variation of instantaneous scale and rotation factors. Further, methodologies can be explored to use it as metadata appropriately.

VI. CONCLUSIONS

We presented two contributions in this work – enhancements to the cross-platform aerial Path-planning dataset and a two-step framework for robust aerial image matching. Our proposed enhancements address several shortcomings in the literature, such as the lack of cross-platform aligned scenes, multiple types of historical galleries, points correspondence, semantic segmentation, etc. The proposed enhanced dataset is very helpful for the evolution of aerial image matching

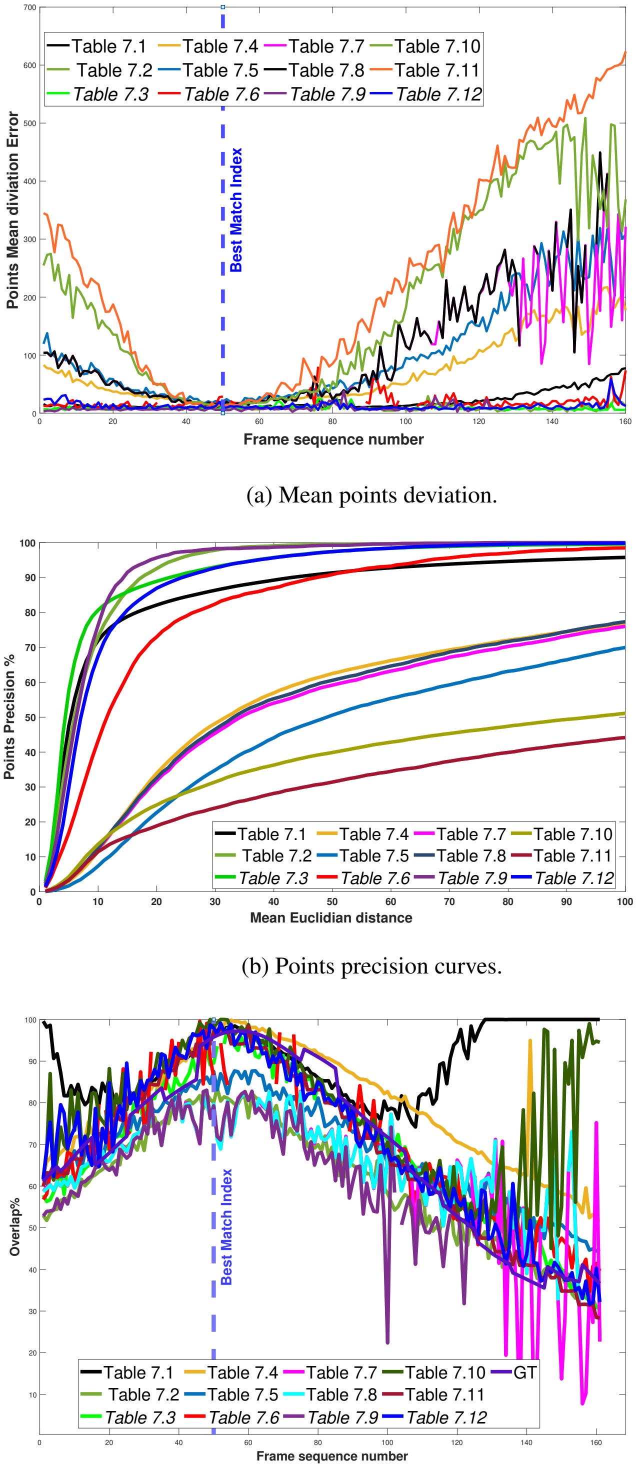


(a) Recall over the gallery.

(b) Precision-Recall curve.

(c) F-score over the gallery

FIGURE 11: Recall, Precision-Recall and F-score curves. Comparison of baseline, proposed fine-tuned without/with extrinsic parameters. The improvement due to the proposed method is clear from all the curves.

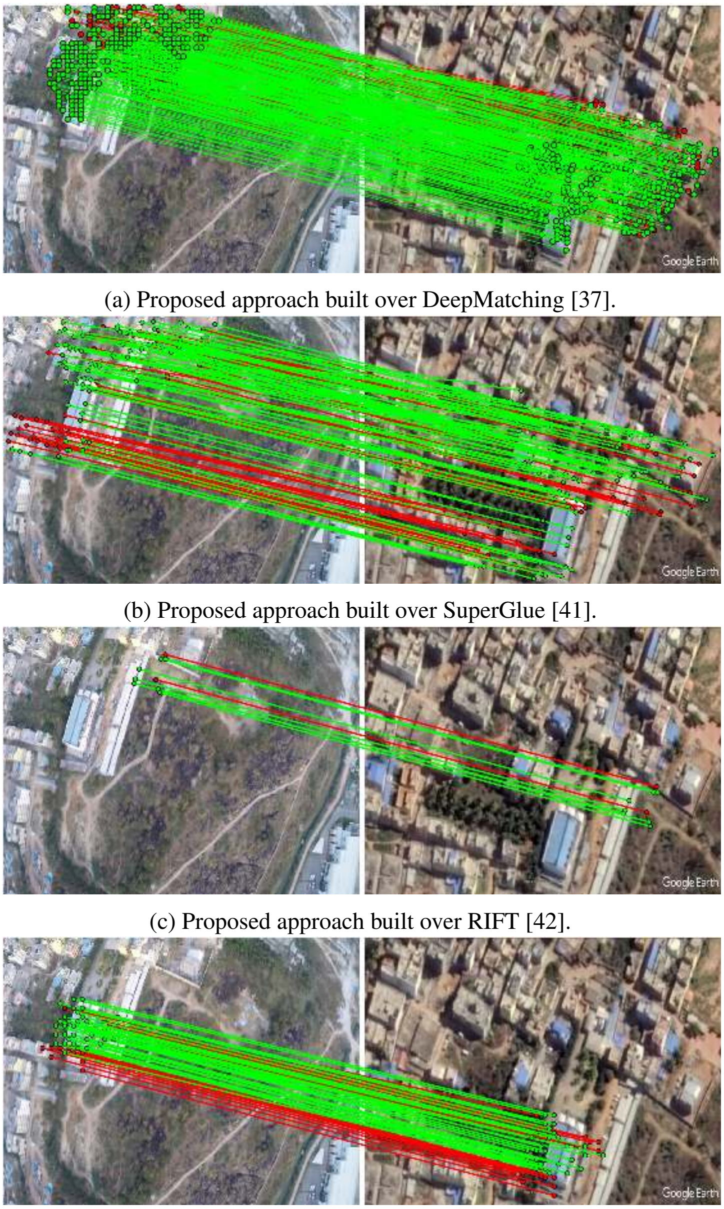


(a) Mean points deviation.

(b) Points precision curves.

(c) Overlap curves.

FIGURE 12: Points performance curves for a DTV query image within an SAT gallery in the same order as table 7. The proposed fine-matching stage improves the overall performance of the considered metrics – MAE, PCK, and Overlap. Italic legend implies the proposed approach. **Best viewed with zoom and color display.**



(a) Proposed approach built over DeepMatching [37].

(b) Proposed approach built over SuperGlue [41].

(c) Proposed approach built over RIFT [42].

(d) Proposed approach built over Patch-NetVLAD [56].

FIGURE 13: An illustration of the fine-matching stage applied to various baseline methods. Green and red lines imply inliers and outliers, respectively. **Best viewed with zoom and color display.**

algorithms. Additionally, we demonstrated a test case of augmenting an open-source aerial dataset for cross-platform classification. It includes a semi-automatic approach to data segregation and enhancing it with cross-platform historical satellite images. We plan to make our enhanced dataset available at https://www.iith.ac.in/~lfovia/downloads.html as part of this publication.

Our two-step framework for robust aerial image matching employs a CNN-based light-weight first step that reduces the load on the fine-matching and helps in tracking the flight path. We developed a methodology for augmenting non-imaging sensor information called metadata or extrinsic parameters. In the second step of the framework, we leverage the *spectral*, *temporal*, and *flow* consistencies followed by cluster analysis for outlier removal for robust matching. We have tested the proposed framework over our dataset, a recent Aerial Template Matching dataset, a synthetic university dataset, and the derived dataset. We have shown efficacy over standard baselines (without and with standard outlier). In summary, we have qualitatively and quantitatively compared our framework against conventional and deep learning-based matching methods and shown that our framework is more effective. We perceive that it is a timely contribution given the increased use of UAVs for a wide variety of applications.

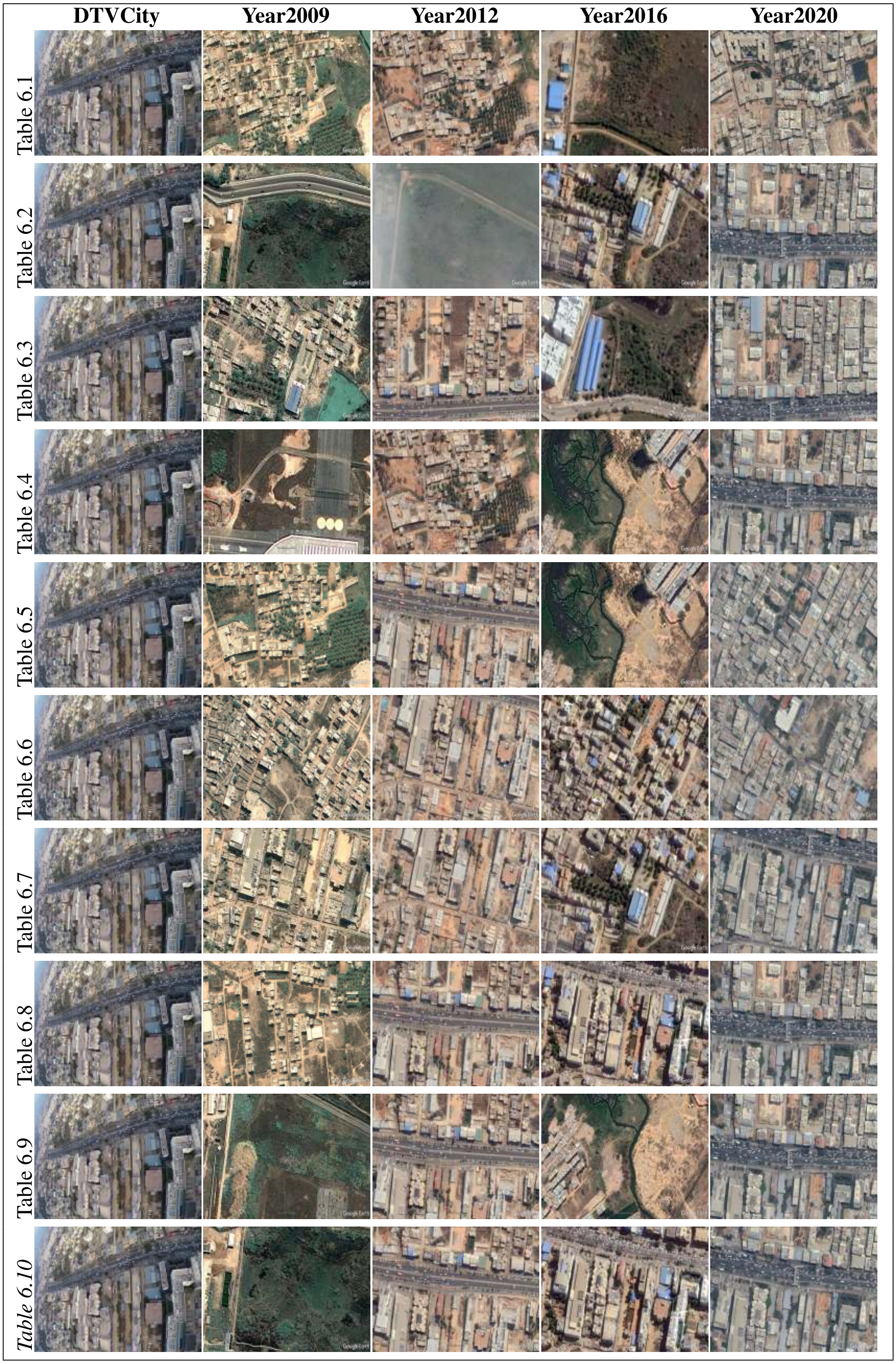


FIGURE 14: Qualitative performance of the coarse-matching approach. Top@1 searches in dataset for *DTVCity* query (round 2). Rows are in the same order as table 6. The last row shows the matching results for the proposed approach. The results in this row are consistently better than most of the other methods in this comparison. **Best viewed with zoom and color display.**

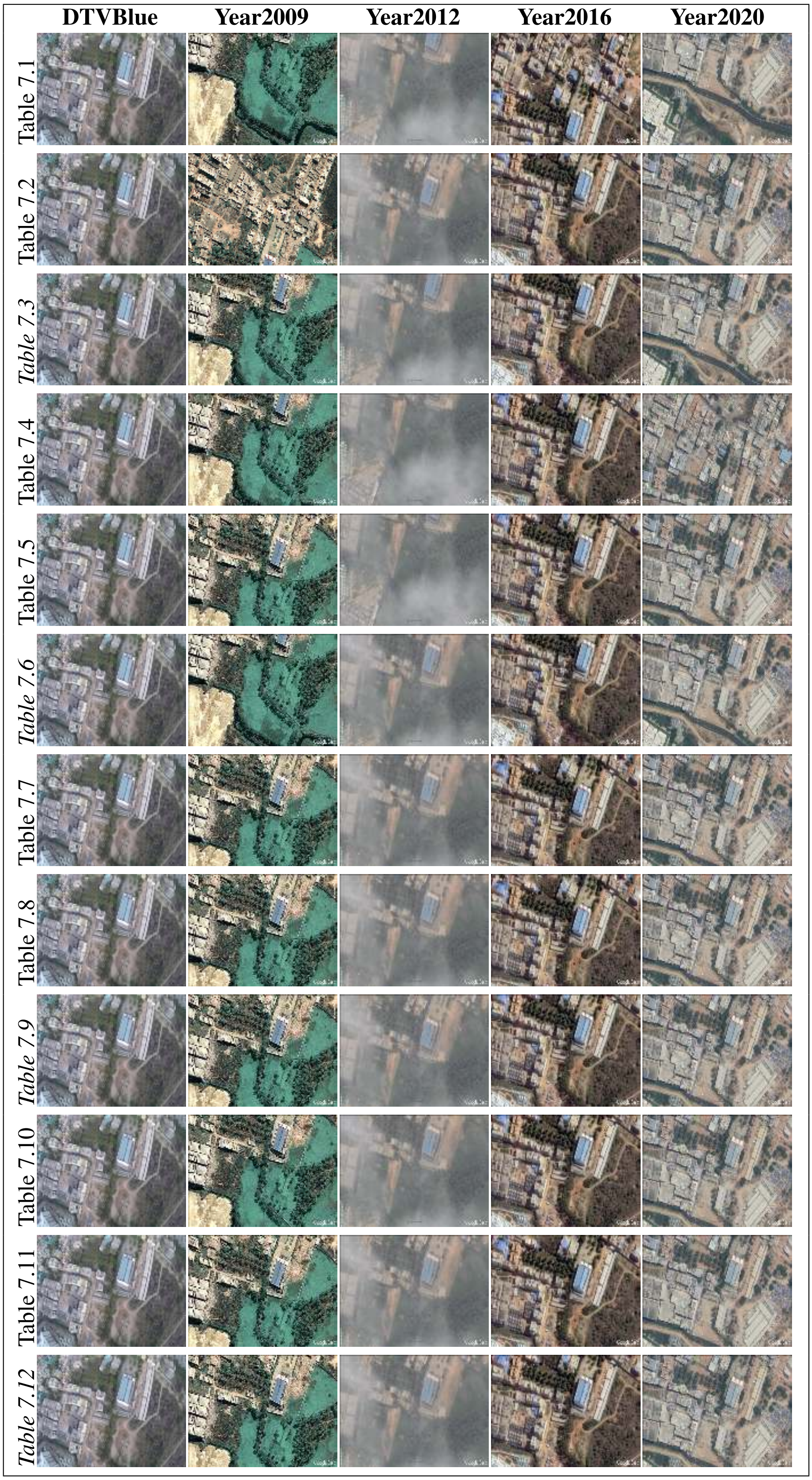


FIGURE 15: Qualitative performance of the fine-matching approach. Top@1 searches in dataset for *DTVBlue* query. Rows are in the same order as table 7. Italic numbered text rows display output of proposed outlier/fine-match approach. **Best viewed with zoom and color display**.

REFERENCES

[1] Z. Liu, H. Wang, L. Weng, and Y. Yang, "Ship rotated bounding box space for ship extraction from high-resolution optical satellite images with complex backgrounds," IEEE Geoscience and Remote Sensing Letters, vol. 13, no. 8, pp. 1074–1078, 2016.

[2] G.-S. Xia, X. Bai, J. Ding, Z. Zhu, S. Belongie, J. Luo, M. Datcu, M. Pelillo, and L. Zhang, "Dota: A large-scale dataset for object detection in aerial images," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3974–3983, 2018.

[3] H. Su, S. Wei, M. Yan, C. Wang, J. Shi, and X. Zhang, "Object detection and instance segmentation in remote sensing imagery based on precise mask R-CNN," in IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium, pp. 1454–1457, IEEE, 2019.

[4] T. Zhang, X. Zhang, X. Ke, X. Zhan, J. Shi, S. Wei, D. Pan, J. Li, H. Su, Y. Zhou, et al., "LS-SSDD-v1. 0: A deep learning dataset dedicated to small ship detection from large-scale sentinel-1 sar images," Remote Sensing, vol. 12, no. 18, p. 2997, 2020.

[5] N. Gorelick, M. Hancher, M. Dixon, S. Ilyushchenko, D. Thau, and R. Moore, "Google earth engine: Planetary-scale geospatial analysis for everyone," Remote Sensing of Environment, vol. 202, pp. 18–27, 2017.

[6] E. Maggiori, Y. Tarabalka, G. Charpiat, and P. Alliez, "Can semantic labeling methods generalize to any city? the inria aerial image labeling benchmark," in 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), pp. 3226–3229, IEEE, 2017.

[7] P. Helber, B. Bischke, A. Dengel, and D. Borth, "Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 12, no. 7, pp. 2217–2226, 2019.

[8] Institute of Computer Graphics and Vision. http://dronedataset.icg.tugraz.at, 2021.

[9] A. L. Majdik, C. Till, and D. Scaramuzza, "The zurich urban micro aerial vehicle dataset," The International Journal of Robotics Research, vol. 36, no. 3, pp. 269–273, 2017.

[10] S. Wang, M. Bai, G. Mattyus, H. Chu, W. Luo, B. Yang, J. Liang, J. Cheverie, S. Fidler, and R. Urtasun, "Torontocity: Seeing the world with a million eyes," arXiv preprint arXiv:1612.00423, 2016.

[11] Y. Tian, C. Chen, and M. Shah, "Cross-view image matching for geo-localization in urban environments," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3608–3616, 2017.

[12] N. Khurshid, T. Hanif, M. Tharani, and M. Taj, "Cross-view image retrieval-ground to aerial image retrieval through deep learning," in International Conference on Neural Information Processing, pp. 210–221, Springer, 2019.

[13] M. T. Chiu, X. Xu, Y. Wei, Z. Huang, A. G. Schwing, R. Brunner, H. Khachatrian, H. Karapetyan, I. Dozier, G. Rose, et al., "Agriculture-vision: A large aerial image database for agricultural pattern analysis," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2828–2838, 2020.

[14] M. Shahid and S. S. Channappayya, "Aerial cross-platform path planning dataset," in Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops, pp. 3936–3945, October 2021.

[15] N. Piasco, D. Sidibé, C. Demonceaux, and V. Gouet-Brunet, "A survey on visual-based localization: On the benefit of heterogeneous data," Pattern Recognition, vol. 74, pp. 90–109, 2018.

[16] M. H. Mughal, M. J. Khokhar, and M. Shahzad, "Assisting uav localization via deep contextual image matching," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 14, pp. 2445–2457, 2021.

[17] Z. Zheng, Y. Wei, and Y. Yang, "University-1652: A Multi-View Multi-Source Benchmark for Drone-Based Geo-Localization," in Proceedings of the 28th ACM International Conference on Multimedia, (New York, NY, USA), p. 1395–1403, Association for Computing Machinery, 2020.

[18] S. Lowry, N. Sünderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, and M. J. Milford, "Visual place recognition: A survey," IEEE Transactions on Robotics, vol. 32, no. 1, pp. 1–19, 2015.

[19] S. Garg, T. Fischer, and M. Milford, "Where is your place, Visual Place Recognition?," arXiv preprint arXiv:2103.06443, 2021.

[20] J. Courbon, Y. Mezouar, N. Guénard, and P. Martinet, "Vision-based navigation of unmanned aerial vehicles," Control Engineering Practice, vol. 18, no. 7, pp. 789–799, 2010.

[21] C. G. Harris and M. Stephens, "A Combined Corner and Edge Detector," in Proceedings of the Alvey Vision Conference, AVC 1988, Manchester, UK, September, 1988 (C. J. Taylor, ed.), pp. 1–6, Alvey Vision Club, 1988.

[22] E. A. R. Martínez, G. Caron, C. Pégard, and D. L. Alabazares, "Photometric path planning for vision-based navigation," in 2020 IEEE International Conference on Robotics and Automation (ICRA), pp. 9007–9013, IEEE, 2020.

[23] J. Shi and Tomasi, "Good features to track," in 1994 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 593–600, 1994.

[24] E. Rosten and T. Drummond, "Fusing points and lines for high performance tracking," in Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1, vol. 2, pp. 1508–1515, Ieee, 2005.

[25] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in 2011 International conference on computer vision, pp. 2564–2571, Ieee, 2011.

[26] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," International Journal of Computer Vision, vol. 60, no. 2, pp. 91–110, 2004.

[27] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," Computer Vision and Image Understanding, vol. 110, no. 3, pp. 346–359, 2008.

[28] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05), vol. 1, pp. 886–893, Ieee, 2005.

[29] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," International journal of computer vision, vol. 42, no. 3, pp. 145–175, 2001.

[30] P. H. Torr and A. Zisserman, "MLESAC: A new robust estimator with application to estimating image geometry," Computer vision and image understanding, vol. 78, no. 1, pp. 138–156, 2000.

[31] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in Computer Vision, IEEE International Conference on, vol. 3, pp. 1470–1470, IEEE Computer Society, 2003.

[32] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in 2007 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8, IEEE, 2007.

[33] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert, "High accuracy optical flow estimation based on a theory for warping," in European Conference on Computer Vision, pp. 25–36, Springer, 2004.

[34] C. Liu, J. Yuen, and A. Torralba, "SIFT Flow: Dense correspondence across scenes and its applications," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 33, no. 5, pp. 978–994, 2010.

[35] J. Kim, C. Liu, F. Sha, and K. Grauman, "Deformable spatial pyramid matching for fast dense correspondences," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2307–2314, 2013.

[36] B. Ham, M. Cho, C. Schmid, and J. Ponce, "Proposal flow: Semantic correspondences from object proposals," IEEE transactions on pattern analysis and machine intelligence, vol. 40, no. 7, pp. 1711–1725, 2017.

[37] J. Revaud, P. Weinzaepfel, Z. Harchaoui, and C. Schmid, "Deepmatching: Hierarchical deformable dense matching," International Journal of Computer Vision, vol. 120, no. 3, pp. 300–323, 2016.

[38] J. Ma, X. Jiang, A. Fan, J. Jiang, and J. Yan, "Image matching from handcrafted to deep features: A survey," International Journal of Computer Vision, vol. 129, no. 1, pp. 23–79, 2021.

[39] K. M. Yi, E. Trulls, V. Lepetit, and P. Fua, "LIFT: Learned Invariant Feature Transform," in Computer Vision – ECCV 2016 (B. Leibe, J. Matas, N. Sebe, and M. Welling, eds.), (Cham), pp. 467–483, Springer International Publishing, 2016.

[40] 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.

[41] P.-E. Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich, "Superglue: Learning feature matching with graph neural networks," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4938–4947, 2020.

[42] J. Li, Q. Hu, and M. Ai, "Rift: Multi-modal image matching based on radiation-variation insensitive feature transform," IEEE Transactions on Image Processing, vol. 29, pp. 3296–3310, 2019.

[43] X. Jiang, J. Ma, and J. Chen, "Progressive filtering for feature matching," in ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 2217–2221, IEEE, 2019.

[44] J. Ma, X. Jiang, J. Jiang, J. Zhao, and X. Guo, "LMR: Learning a two-class classifier for mismatch removal," IEEE Transactions on Image Processing, vol. 28, no. 8, pp. 4045–4059, 2019.

[45] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in Advances in Neural Information Processing Systems (C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, eds.), vol. 28, Curran Associates, Inc., 2015.

[46] N. Garcia and G. Vogiatzis, "Learning non-metric visual similarity for image retrieval," Image and Vision Computing, vol. 82, pp. 18–25, 2019.

[47] S. Basu, S. Ganguly, S. Mukhopadhyay, R. DiBiano, M. Karki, and R. Nemani, "Deepsat: A learning framework for satellite imagery," in Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems, SIGSPATIAL '15, (New York, NY, USA), Association for Computing Machinery, 2015.

[48] T. Weyand, I. Kostrikov, and J. Philbin, "Planet-photo geolocation with convolutional neural networks," in European Conference on Computer Vision, pp. 37–55, Springer, 2016.

[49] Z. Yang, T. Dan, and Y. Yang, "Multi-temporal remote sensing image registration using deep convolutional features," IEEE Access, vol. 6, pp. 38544–38555, 2018.

[50] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556, 2014.

[51] I. Rocco, M. Cimpoi, R. Arandjelović, A. Torii, T. Pajdla, and J. Sivic, "Neighbourhood consensus networks," arXiv preprint arXiv:1810.10510, 2018.

[52] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778, 2016.

[53] B. Ham, M. Cho, C. Schmid, and J. Ponce, "Proposal flow: Semantic correspondences from object proposals," IEEE transactions on pattern analysis and machine intelligence, vol. 40, no. 7, pp. 1711–1725, 2017.

[54] X. Jiang, J. Ma, G. Xiao, Z. Shao, and X. Guo, "A review of multimodal image matching: Methods and applications," Information Fusion, vol. 73, pp. 22–71, 2021.

[55] B. Kong, J. Supancic, D. Ramanan, and C. C. Fowlkes, "Cross-domain image matching with deep feature maps," International Journal of Computer Vision, vol. 127, no. 11-12, pp. 1738–1750, 2019.

[56] S. Hausler, S. Garg, M. Xu, M. Milford, and T. Fischer, "Patch-netvlad: Multi-scale fusion of locally-global descriptors for place recognition," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 14141–14152, 2021.

[57] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "NetVLAD: CNN architecture for weakly supervised place recognition," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2016.

[58] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," Communications of the ACM, vol. 24, no. 6, pp. 381–395, 1981.

[66] M. Mueller, N. Smith, and B. Ghanem, "A benchmark and simulator for uav tracking," in European Conference on Computer Vision, pp. 445–461, Springer, 2016.

[59] Y. Djenouri and J. Hjelmervik, "Hybrid Decomposition Convolution Neural Network and Vocabulary Forest for Image Retrieval," in 2020 25th International Conference on Pattern Recognition (ICPR), pp. 3064–3070, IEEE, 2021.

[60] Y. Djenouri, J. Hatleskog, J. Hjelmervik, E. Bjorne, T. Utstumo, and M. Mobarhan, "Deep learning based decomposition for visual navigation in industrial platforms," Applied Intelligence, pp. 1–17, 2021.

[61] Y. Bai, W. Yu, T. Xiao, C. Xu, K. Yang, W.-Y. Ma, and T. Zhao, "Bag-of-words based deep neural network for image retrieval," in Proceedings of the 22nd ACM International Conference on Multimedia, MM '14, (New York, NY, USA), p. 229–232, Association for Computing Machinery, 2014.

[62] X. Yang, X. Gao, B. Song, and B. Han, "Hierarchical deep embedding for aurora image retrieval," IEEE Transactions on Cybernetics, vol. 51, no. 12, pp. 5773–5785, 2021.

[63] N. Sünderhauf, S. Shirazi, F. Dayoub, B. Upcroft, and M. Milford, "On the performance of convnet features for place recognition," in 2015 IEEE/RSJ international conference on intelligent robots and systems (IROS), pp. 4297–4304, IEEE, 2015.

[64] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," IEEE Transactions on Image Processing, vol. 13, no. 4, pp. 600–612, 2004.

[65] S. M. Lowry, M. J. Milford, and G. F. Wyeth, "Transforming morning to afternoon using linear regression techniques," in 2014 IEEE International Conference on Robotics and Automation (ICRA), pp. 3950–3955, 2014.

[67] A. Rau, G. Garcia-Hernando, D. Stoyanov, G. J. Brostow, and D. Turmukhambetov, "Predicting Visual Overlap of Images Through Interpretable Non-Metric Box Embeddings," in European Conference on Computer Vision, pp. 629–646, Springer, 2020.

[68] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid, "Deepflow: Large displacement optical flow with deep matching," in Proceedings of the IEEE International Conference on Computer Vision, pp. 1385–1392, 2013.

[69] E. McCreath et al., "Partial matching of planar polygons under translation and rotation," Conference Organising Committee, 2008.

[70] E. Arkin, L. Chew, D. Huttenlocher, K. Kedem, and J. Mitchell, "An efficiently computable metric for comparing polygonal shapes," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 13, no. 3, pp. 209–216, 1991.

[71] N. Ufer and B. Ommer, "Deep semantic feature matching," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6914–6923, 2017.

[72] J. Zhang, D. Sun, Z. Luo, A. Yao, L. Zhou, T. Shen, Y. Chen, L. Quan, and H. Liao, "Learning two-view correspondences and geometry using order-aware network," in Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), October 2019.