DEEP LEARNING BASED LANDMARK MATCHING FOR AERIAL GEOLOCALIZATION

Koundinya Nouduri[1], Filiz Bunyak[1], Shizeng Yao[1], Hadi Aliakbarpour[1], Sanjeev Agarwal[2], Raghuveer Rao[3], Kannappan Palaniappan[1,]*

[1] Department of EECS, University of Missouri-Columbia, MO, USA
[2] U.S. Army CCDC C5ISR Center, USA
[3] U.S. Army CCDC Army Research Laboratory, USA

ABSTRACT

Visual odometry has gained increasing attention due to the proliferation of unmanned aerial vehicles, self-driving cars, and other autonomous robotics systems. Landmark detection and matching are critical for visual localization. While current methods rely upon point-based image features or descriptor mappings we consider landmarks at the object level. In this paper, we propose LMNet a deep learning based landmark matching pipeline for city-scale, aerial images of urban scenes. LMNet consists of a Siamese network, extended with a multi-patch based matching scheme, to handle off-center landmarks, varying landmark scales, and occlusions of surrounding structures. While there exist a number of landmark recognition benchmark datasets for ground-based and nadir aerial or satellite imagery, there is a lack of datasets and results for oblique aerial imagery. We use a unique *unsupervised* multi-view landmark image generation pipeline for training and testing the proposed matching pipeline using over 0.5 million real landmark patches. Results for aerial landmark matching across four cities show promising results.

Index Terms— Landmark matching, aerial surveillance, deep learning, geolocalization, visual SLAM

1. INTRODUCTION

Visual localization using foundational geospatial data is a critical component for place recognition, navigation, and loop closure or path recovery in environments where sensor metadata may be noisy and intermittent. With the recent proliferation of unmanned aerial vehicles, self-driving cars, and other autonomous robotics systems, there is a growing need for visual localization capabilities. Visual localization [1] aims to identify a query image within a geo-referenced collection that can consist of a single image, a set of images, or a 2D/3D model of the scene and enables robust operation even when platform and sensor metadata are noisy or sporadic. Instead of the typical low-level image feature-based approach we consider an object level landmark-based method for visual localization. Landmark matching in urban scenarios, however, is challenging due to non-distinctive, repetitive structures such as rooftops of houses or windows of tall buildings; scene clutter; varying sized landmarks; appearance changes caused by differences in time of day, in viewing direction, occlusions and view-dependent uncovering of different structures surrounding tall buildings. Landmark identification is even more difficult in *oblique* aerial imagery as tall landmarks undergo a high degree of perspective change between widely separated directional views. Until the recent advances in deep learning, image matching was predominantly performed by feature point detection and matching using carefully designed feature detectors and descriptors [2]. While these classical approaches produce satisfactory results for image pairs captured from similar viewing directions, their performances severely deteriorate with larger viewing angles (nadir vs oblique) or with repetitive structures. Inspired by the multi-level abstraction capabilities of deep learning methods and their recent success in matching [3, 4, 5, 6, 7, 8, 9, 10] , we have developed a novel object level landmark matching pipeline suitable for visual odometry, with the core consisting of a single-patch Siamese network [11, 12]. In order to handle varying scale of landmarks and occlusions from surrounding structures, we extended the single-patch pipeline to a multi-patch ensemble. Based on our work in bundle adjustment [13, 14, 15] and city-scale 3D reconstruction [16, 17], we also developed a 3D enabled, *unsupervised*, training data generation pipeline.

2. DEEP LEARNING-BASED LANDMARK MATCHING USING LMNET

We have designed a deep learning-based landmark matching pipeline for city-scale, aerial images of urban scenes. Our LMNet deep architecture consists of a Siamese network with a ResNet [18] feature extraction backbone described in §2.2. In order to handle multi-scale landmarks and occlusions from surrounding background structures, we extended our single-

[*] This work was partially supported by awards from U.S. Army Research Laboratory W911NF-1820285, Army Research Office DURIP W911NF-1910181, and U.S. Air Force Research Laboratory FA8750-19-2-0001. Public release approval 17848. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the U. S. Government or agency thereof.

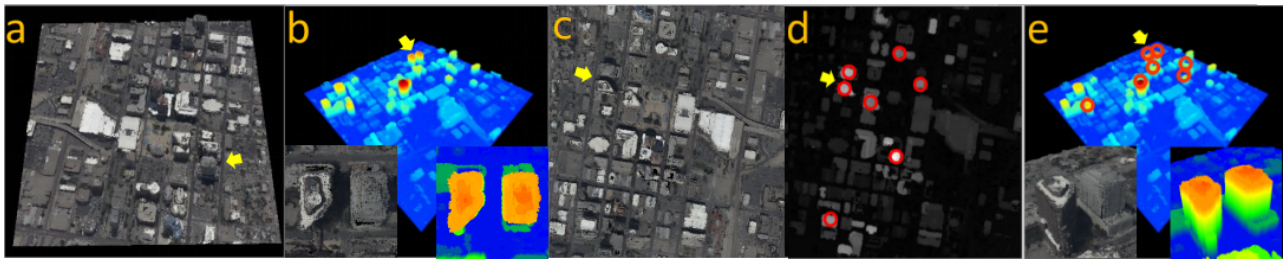

Fig. 1: 3D enabled multi-view landmark image generation: (a) 3D city-scale reconstruction using ABQ WAMI dataset; (b) rendered 3D point cloud (blue to red for low to high elevations); (c) orthographic view of RGB point cloud; (d) tall buildings automatically identified as landmarks of interest in orthographic view (grayscale height map); and, (e) two tall buildings visualized using MU Nimbus2 3D software.

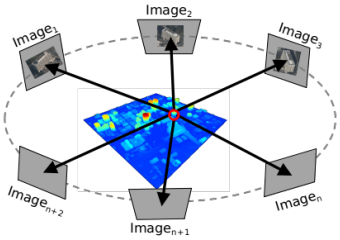

Fig. 2: Forward projection from 3D real-world coordinates to high-resolution 2D image sequence for different camera viewing directions using bundle adjusted camera poses.

patch pipeline to a multi-patch ensemble. The pipeline was trained with a set of matching and non-matching image pairs. Unsupervised aerial landmark training data from several cities were generated in a scalable manner as described next.

2.1. 3D enabled multi-view landmark image generation

Success of supervised deep learning approaches critically depends upon the availability of large amounts of labeled training data. Such datasets are lacking for aerial landmark matching. Based on our prior work in bundle adjustment [13, 14, 15] and city-scale 3D reconstruction [16, 17], we have developed an *unsupervised* training data generation pipeline for landmark matching. Major steps of this pipeline are shown in Algorithm 1 and illustrated in Figures 1 and 2. To generate the training and testing data for our proposed landmark matching pipeline, we used wide-area motion imagery (WAMI) [19] data with known camera poses and corresponding 3D point clouds. These WAMI data were captured using high-resolution visible RGB cameras, together with consumer-level quality metadata from noisy GPS and IMU sensors associated with each camera view that are corrected using our Structure-from-Motion and bundle adjustment algorithm, called *BA4S* [13]. BA4S receives the image sequences as input along with the noisy camera position vectors (GPS) and rotation angles (IMU). It uses a robust and very fast method to optimize the camera poses (the positions and orientations) by using a non-linear least-squares solver (Levenberg-Marquardt) algorithm together with a robust function which is able to efficiently handle strong parallaxes [20] induced by tall 3D structures and building in WAMI scenarios. Having the camera poses optimized, dense city-scale 3D point clouds were reconstructed using the multi-view WAMI [16]. A reconstructed city-scale model is shown in Figure 1. Using the 3D point cloud reconstruction, a height map (Figure 1b) and an orthographic view (Figure 1c) of the urban area scene were constructed. Tall structures (landmarks) were automatically located by thresholding the height map and are marked by red circles in Figures 1d and 1e. The spatial positions of these tall structures combined with their corresponding heights constitute their 3D real-world coordinates. To obtain different views of the same landmark, the 3D coordinates of the landmark rooftops were forward projected to the high-resolution 2D sequence of images (Figure 2) using bundle adjusted camera poses, comprised of the rotation matrix $\mathbf{R}_i$ and translation vector $\mathbf{t}_i$ for the i^{th} camera or view,

$$\mathbf{x}_j = \mathbf{K}\,[\mathbf{R}_i|\mathbf{t}_i]\,\mathbf{X}_j \tag{1}$$

where $\mathbf{X}_j$ is the 3D (homogeneous) coordinate of the j^{th} landmark, $\mathbf{x}_j$ is its associated 2D pixel (homogeneous) coordinate in the i^{th} camera view and $\mathbf{K}$ is the camera intrinsic matrix assumed to be the same for all views.

Algorithm 1 *3D enabled multi-view landmark image generation for unsupervised training data*

1. Bundle adjustment [13, 14, 15] (Fig. 1a)
2. City-scale 3D model reconstruction [16, 17] (Fig. 1b)
3. City-scale orthographic view and height map generation from 3D model (Fig. 1c-d)
4. Automatic landmark detection from height map (Fig. 1e)
5. Identification of landmark 3D real-world coordinates
6. Forward projection: 3D coordinates to high-resolution 2D image sequence using bundle adjusted camera poses (Fig. 2)
7. Visibility check: ray-tracing + 3D local neighborhood search

2.2. Landmark matching using a Siamese network

The proposed pipeline uses a Siamese network to infer whether a given pair of image patches correspond to the

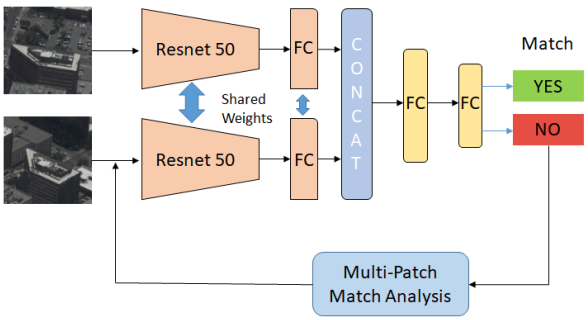

Fig. 3: Proposed LMNet multi-view landmark matching architecture consisting of a deep learning Siamese matching network and a multi-patch match analysis module that compensates for center-offset, scale difference and occlusions of the landmark of interest.

same landmark or not. The network consists of two parts, image patch feature extraction streams that share weights followed by a binary classifier. Feature extraction uses two pre-trained ResNet50 architectures, whose outputs are connected to a pair of fully connected (FC) layers. These outputs are vertically concatenated, then projected to two fully connected layers, to decide on match versus no-match binary classification. The network terminates with a sigmoid layer and is minimized using binary cross entropy loss,

$$L(y,p) = -\sum_{i} (y_i \log(p_i) + (1-y_i)\log(1-p_i)) \quad (2)$$

where y_i is the binary indicator for the predicted class label (1 for incorrect classification) and p_i is the predicted probability for a matching patch summed over all landmark patches in the mini-batch. *Matching pairs* were selected from different views of the same landmark using 3D enabled forward projection as described in Algorithm 1 and shown in Figure 2. Multiple matching pairs were created for each landmark in an unsupervised manner. Matching pairs were grouped based on the camera viewing direction angle differences. Three clusters were created for angles differences of (0°- 15°], (15°- 30°], and (30°- 45°]. *Non-matching pairs* are created by pairing views of different landmarks from the same city.

2.3. Multi-patch matching scheme

Matching landmarks across different views is a challenging task. Different views not only result in appearance changes in the landmark of interest, but also in the surrounding background structures. In order to reduce the adverse effects on matching performance of: (1) off-center landmarks, (2) small landmarks, and (3) occlusions due to surrounding structures, we developed a multi-patch matching scheme. This scheme aims to reduce false-negatives (mis-classification of matching pairs) and is activated only when the network output is the non-matching class. The proposed multi-patch matching scheme then extracts nine 300×300 neighboring sub-patches from the query and reference image patches as shown in Figure 4. The sub-patches are resized and fed to the Siamese network (Section 2.2) to generate 9×9 comparisons. A pair of landmark images is considered a match when K or more tests (out of 81) are of the matching class.

Table 1: Characteristics of aerial imagery. BK landmarks were used only for testing. Height threshold for building masks was 75.

City	Number of views	Angle btw views (degree)	# Training landmarks	# Testing landmarks
LA	351	0.975	28	13
ABQ	429	1.190	27	9
SYR	295	0.819	19	9
BK	220	0.611	0	34

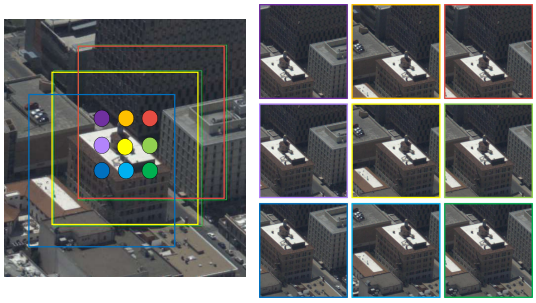

Fig. 4: Multi-patch matching scheme. Left: Original image patch with locations of 9 sub-patches marked as colored dots and 3 patches shown. Right: Nine cropped sub-patches centered on colored dots.

2.4. Estimation of inter-view angles

Two variations of the LMNet in Figure 3 were developed to estimate the viewing angle differences between matching landmark image patches. The problem can be formulated as a classification or regression problem. Several modifications were made to convert the LMNet landmark matching network described in §2.2 to an angle estimation network. In the regression-based extension, the final network layer was modified to produce a single output optimized using a mean-squared error loss function. In the classification-based extension the angle between views was binned into three classes: (0°, 10°], (10°, 20°], and (20°, 30°]. The final network layer was modified to output 3 classes using softmax and optimized using the cross-entropy loss function. See Table 3 for results.

3. EXPERIMENTAL RESULTS

WAMI data was collected by Transparent Sky, LLC using an aircraft with on-board GPS and IMU measurements in a circular flight pattern. Table 1 summarizes the dataset information. Landmark patch sizes are resized to 500×500 pixels. The proposed pipeline was trained with landmarks from LA, ABQ, SYR, then tested on *separate* landmarks from LA,

Table 2: Single-patch and multi-patch landmark matching accuracies. TN & TP refer to true negative and positive rates. Results organized by aerial dataset, non-matching & matching image pairs, and angle between views. Multi-patch results use a matching threshold of $K = 40$.

	Aerial Dataset	Non-Matching Image Pairs - (TN)			Matching Image Pairs (TP)		
	Angle between views	(0°,15°]	(15°,30°]	(30°,45°]	(0°,15°]	(15°,30°]	(30°,45°]
Single-Patch	Albuquerque, NM	94.03%	94.11%	94.11%	98.17%	81.79%	70.00%
	Los Angeles, CA	81.90%	81.83%	81.70%	98.95%	89.30%	79.52%
	Syracuse, NY	93.91%	93.82%	94.15%	98.13%	80.78%	70.08%
	Berkeley, CA	91.41%	91.31%	91.44%	98.46%	80.82%	66.05%
Multi-Patch	Albuquerque, NM	93.89%	93.93%	89.24%	98.24%	82.32%	87.02%
	Los Angeles, CA	76.26%	76.21%	57.30%	99.12%	92.29%	95.86%
	Syracuse, NY	93.44%	93.40%	87.78%	98.99%	89.30%	95.74%
	Berkeley, CA	90.05%	89.92%	79.78%	98.85%	85.15%	91.69%

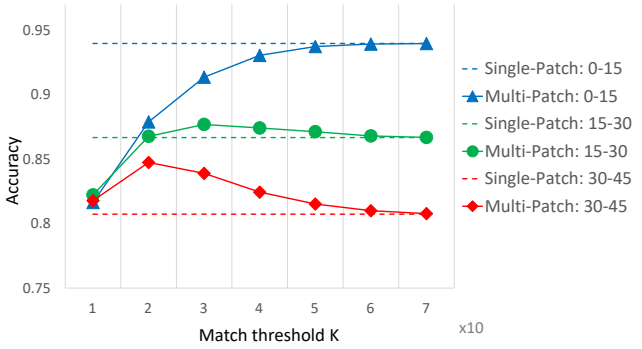

Fig. 5: Effect of multi-patch match threshold K on match accuracy for multi-patch landmark matching compared to single-patch results (horizontal lines since thresholding does not apply).

ABQ, SYR, BK datasets. BK was the held out dataset and was not used for training. In total, 142,272 matching and 143,627 non-matching image patch pairs were used for training. Another 262,212 image patch pairs were used for testing.

Table 2 shows landmark matching accuracies for single-patch and multi-patch matching schemes. Results on unseen BK dataset demonstrates the generalization capabilities of the landmark matching deep network. When the global single-patch scheme is used, correct classification of non-matching image pairs is not affected by the viewing direction. However, classification accuracies for matching image pairs (TP) decreases significantly by 27.0% (averaged over four cities) for large viewing angle differences of 30° to 45°, compared to narrow viewing angles of 0° to 15°, due to large perspective appearance changes between the views. Using the multi-patch matching scheme, accuracy improves significantly for large viewing differences of 30° to 45°, resulting in only a 6.2% decrease in accuracy for matching image pairs (TP), combined with a 9.9% decrease in accuracy for non-matching pairs (TN), due to false-positives introduced by multi-patch matching. Overall multi-patch landmark matching provides a net benefit. Figure 5 shows the difference in performance between single- and multi-patch matching methods and the influence of threshold K on the latter. For large K accuracy converges to the global single-patch matching scheme. For low values of K, matching accuracies for viewing directions 0° to 15°decrease due to increased false-positives. The highest matching accuracy was obtained using a threshold for $K \in [20, 40]$ which corresponds to 25% to 50% of the 81 comparisons between two sets of 3×3 sub-patches (see Figure 4). For these cases, false-negatives decrease for viewing directions of 30° to 45°. When viewing directions can be estimated, the match threshold K can be set to optimize performance. Table 3 shows accuracy for viewing direction estimation. The regression network has a 92% estimation accuracy which outperforms the classification based method by 18%.

Table 3: View separation angle estimation across different cities. Angle estimation error threshold for regression is set to $\pm 5°$.

	LA	ABQ	SYR	BK	Overall
Regression	91.49	98.12	83.53	94.08	92.60
Classification	71.34	81.50	68.28	75.75	74.57

4. CONCLUSIONS

In this paper, we proposed a deep learning based landmark matching pipeline, LMNet, for city-scale, aerial images of urban scenes. LMNet is composed of a Siamese network with a ResNet feature extraction sub-network, with a novel multi-patch matching scheme used to handle off-center landmarks, varying scale of landmarks, and occlusion from surrounding structures. LMNet uses multi-level abstraction capabilities of deep learning methods to capture regional scene information using multi-patch image sets for robust landmark matching. LMNet is trained in an unsupervised manner using our 3D-enabled multi-view landmark image patch generation, which allowed us to generate a large number of matching multi-view image patches *without* manual labeling. Experiments on data from four cities, Los Angeles, Albuquerque, Syracuse, and Berkeley show an average recognition accuracy of nearly 90% for aerial landmark matching. The output of this method can be used to initialize a 3D-2D (PnP) or 2D-2D camera pose estimation algorithms within a visual navigation framework.

5. REFERENCES

[1] Nathan Piasco, Desire Sidibé, Cedric Demonceaux, and Valerie Gouet-Brunet, "A survey on visual-based localization: On the benefit of heterogeneous data," *Pattern Recognition*, vol. 74, pp. 90–109, 2018.

[2] Chendcai Leng, Hai Zhang, Bo Li, Guorong Cai, Zhao Pei, and Li He, "Local feature descriptor for image matching: A survey," *IEEE Access*, vol. 7, pp. 6424–6434, 2019.

[3] Hyungtae Lee and Heesung Kwon, "Going deeper with contextual CNN for hyperspectral image classification," *IEEE Transactions on Image Processing*, vol. 26, no. 10, pp. 4843–4855, 2017.

[4] Priya Narayanan, Christoph Borel-Donohue, Hyungtae Lee, Heesung Kwon, and Raghuveer M Rao, "A real-time object detection framework for aerial imagery using deep neural networks and synthetic training images," in *SPIE Signal Processing, Sensor/Information Fusion, and Target Recognition XXVII*, 2018, vol. 10646, p. 1064614.

[5] Edgar Simo-Serra, Eduard Trulls, Luis Ferraz, Iasonas Kokkinos, Pascal Fua, and Francesc Moreno-Noguer, "Discriminative learning of deep convolutional feature point descriptors," in *IEEE Int. Conference on Computer Vision*, 2015.

[6] Hani Altwaijry, Eduard Trulls, James Hays, Pascal Fua, and Serge Belongie, "Learning to match aerial images with deep attentive architectures," in *IEEE Conf. Computer Vision and Pattern Recognition*, 2016.

[7] Iaroslav Melekhov, Juho Kannala, and Esa Rahtu, "Siamese network features for image matching," in *Int. Conf. Pattern Recognition*, 2016, pp. 378–383.

[8] Andy Zeng, Shuran Song, Matthias Nießner, Matthew Fisher, Jianxiong Xiao, and Thomas Funkhouser, "3DMatch: Learning local geometric descriptors from RGB-D reconstructions," in *Proc. IEEE Conf. Computer Vision Pattern Recognition*, 2017.

[9] Yuki Ono, Eduard Trulls, Pascal Fua, and Kwang Moo Yi, "LF-Net: Learning local features from images," in *Advances in Neural Information Processing Systems*, 2018, pp. 6234–6244.

[10] Mihai Dusmanu, Ignacio Rocco, Tomas Pajdla, Marc Pollefeys, Josef Sivic, Akihiko Torii, and Torsten Sattler, "D2-net: A trainable cnn for joint description and detection of local features," in *IEEE Conf. Computer Vision and Pattern Recognition*, 2019, pp. 8092–8101.

[11] Sergey Zagoruyko and Nikos Komodakis, "Learning to compare image patches via convolutional neural networks," in *IEEE Conf. Computer Vision and Pattern Recognition*, 2015, pp. 4353–4361.

[12] Ejaz Ahmed, Michael J. Jones, and Tim K. Marks, "An improved deep learning architecture for person re-identification.," in *IEEE Conf. Computer Vision and Pattern Recognition*, 2015, pp. 3908–3916.

[13] Hadi AliAkbarpour, Kannappan Palaniappan, and Guna Seetharaman, "Fast structure from motion for sequential and wide area motion imagery," in *Int. Conf. Computer Vision Workshop*, Dec 2015, pp. 1086–1093.

[14] Abdullah Akay, Hadi AliAkbarpour, Kannappan Palaniappan, and Guna Seetharaman, "Camera auto-calibration for planar aerial imagery, supported by camera metadata," in *IEEE Applied Imagery Pattern Recognition Workshop*, Oct 2017, pp. 1–5.

[15] Hadi Aliakbarpour, Kannappan Palaniappan, and Guna Seetharaman, "Robust camera pose refinement and rapid SfM for multiview aerial imagery—without RANSAC," *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 11, pp. 2203–2207, Nov 2015.

[16] Shizeng Yao, Hadi AliAkbarpour, Guna Seetharaman, and Kannappan Palaniappan, "3D patch-based multi-view stereo for high-resolution imagery," in *SPIE Geospatial Informatics, Motion Imagery, and Network Analytics VIII*, 2018, vol. 10645, pp. 146 – 153.

[17] Hadi Aliakbarpour, Joao F. Ferreira, VB. Surya Prasath, Kannappan Palaniappan, Guna Seetharaman, and Jorge Dias, "A probabilistic framework for 3D reconstruction using heterogeneous sensors," *IEEE Sensors*, vol. 17, no. 9, pp. 1–2, May 2017.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *IEEE Conf. Computer Vision Pattern Recognition*, 2016, pp. 770–778.

[19] Kannappan Palaniappan, Raghuveer M Rao, and Guna Seetharaman, "Wide-area persistent airborne video: Architecture and challenges," in *Distributed Video Sensor Networks*, pp. 349–371. Springer, 2011.

[20] Hadi AliAkbarpour, Gunasekaran Seetharaman, and Kannappan Palaniappan, "Method for fast camera pose refinement for wide area motion imagery," Oct. 8 2019, US Patent 10,438,366.