

# Identifying Corresponding Patches in SAR and Optical Images With a Pseudo-Siamese CNN

Lloyd H. Hughes, *Student Member, IEEE*, Michael Schmitt<sup>✉</sup>, *Senior Member, IEEE*, Lichao Mou, Yuanyuan Wang<sup>✉</sup>, *Member, IEEE*, and Xiao Xiang Zhu<sup>✉</sup>, *Senior Member, IEEE*

**Abstract**—In this letter, we propose a pseudo-siamese convolutional neural network architecture that enables to solve the task of identifying corresponding patches in very high-resolution optical and synthetic aperture radar (SAR) remote sensing imagery. Using eight convolutional layers each in two parallel network streams, a fully connected layer for the fusion of the features learned in each stream, and a loss function based on binary cross entropy, we achieve a one-hot indication if two patches correspond or not. The network is trained and tested on an automatically generated data set that is based on a deterministic alignment of SAR and optical imagery via previously reconstructed and subsequently coregistered 3-D point clouds. The satellite images, from which the patches comprising our data set are extracted, show a complex urban scene containing many elevated objects (i.e., buildings), thus providing one of the most difficult experimental environments. The achieved results show that the network is able to predict corresponding patches with high accuracy, thus indicating great potential for further development toward a generalized multisensor key-point matching procedure.

**Index Terms**—Convolutional neural networks (CNNs), data fusion, deep learning, deep matching, image matching, optical imagery, synthetic aperture radar (SAR).

## I. INTRODUCTION

THE identification of corresponding image patches is used extensively in computer vision and remote sensing-related image analysis, especially in the framework of stereoapplications or coregistration issues. While many successful hand-crafted approaches, specifically designed for the matching of optical images, exist [1], to this date, the matching of images acquired by different sensors still remains a widely unsolved challenge [2]. This particularly holds for a joint exploitation of synthetic aperture radar (SAR) and optical imagery caused



Fig. 1. Simple detached multistory building as (Left) SAR amplitude image and (Right) optical photograph.

by two completely different sensing modalities: SAR imagery collects information about the physical properties of the scene and follows a range-based imaging geometry, while optical imagery reflects the chemical characteristics of the scene and follows a perspective imaging geometry. Hence, structures elevated above the ground level, such as buildings or trees, show strongly different appearances in both SAR and optical images (see Fig. 1), in particular when dealing with very high-resolution (VHR) data.

In order to deal with the problem of multisensor key-point matching, several sophisticated approaches have been proposed, e.g., exploiting phase congruency as a generalization of gradient information [3]. However, even sophisticated hand-crafted descriptors reach their limitations for highly resolving data showing densely built-up urban scenes, which—in the SAR case—is often difficult to interpret even for trained experts.

Therefore, this letter aims at learning a multisensor correspondence predictor for SAR and optical image patches of the state-of-the-art VHR data. Inspired by promising results achieved in the context of stereomatching for optical imagery [4], [5], we also make use of a convolutional neural network (CNN). The major difference of this letter to these purely optical approaches is that we focus on the aforementioned, distinctly more complicated multisensor setup and, therefore, design a specific pseudo-siamese network architecture with two separate, yet identical convolutional streams for processing SAR and optical patches in parallel instead of a weight-shared siamese network in order to deal with the heterogeneous nature of the input imagery.

## II. NETWORK ARCHITECTURE

### A. Pseudo-Siamese Convolutional Network

Since SAR and optical images lie on different manifolds, it is not advisable to compare them directly by descriptors

Manuscript received June 30, 2017; revised September 18, 2017, October 30, 2017, and December 19, 2017; accepted January 24, 2018. Date of publication March 12, 2018; date of current version April 20, 2018. This work was supported in part by the China Scholarship Council, in part by the European Research Council under the EU Horizon 2020 Research and Innovation Program under Grant ERC-2016-StG-714087(So2Sat), in part by the Helmholtz Association under the framework of the Young Investigators Group SiPEO under Grant VH-NG-1018, and in part by the German Research Foundation under Grant SCHM 3322/1-1. (Corresponding author: Xiao Xiang Zhu.)

L. H. Hughes, M. Schmitt, L. Mou, and Y. Wang are with Signal Processing in Earth Observation, Technical University of Munich, 80333 Munich, Germany (e-mail: lloyd.hughes@tum.de; m.schmitt@tum.de; lichao.mou@dlr.de; yuanyuan.wang@dlr.de).

X. X. Zhu is with Signal Processing in Earth Observation, Technical University of Munich, 80333 Munich, Germany, and also with the Remote Sensing Technology Institute, German Aerospace Center, 82234 Wessling, Germany (e-mail: xiao.zhu@dlr.de).

Color versions of one or more of the figures in this letter are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/LGRS.2018.2799232

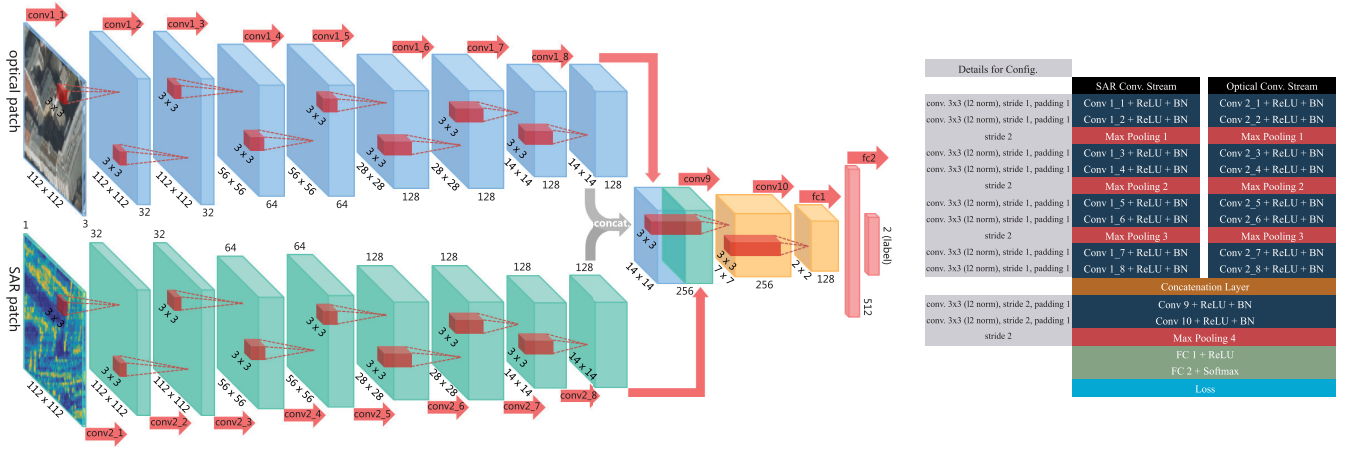


Fig. 2. Proposed pseudo-siamese network architecture and layer configuration.

designed for matching optical patches. Siamese CNN architectures are not suitable for this task either, as weights are shared between the parallel streams, thus implying the inputs share similar image features. In order to cope with the strongly different **geometric and radiometric** appearances of SAR and optical imagery, in [6], we proposed a pseudo-siamese network architecture with **two separate**, yet identical convolutional streams, which process the SAR patch and the optical patch in parallel and only **fuse** the resulting information at a later decision stage. Using this architecture, **the network is constrained to first learn meaningful representations of the input SAR patch and the optical patch separately and to combine them on a higher level**. The work presented in this letter is an extension of [6] by improving the fusion part of the network architecture, using a different training strategy, and resorting to nonlocally prefiltered SAR patches instead of temporal mean maps. In addition, we evaluate the network on a **deterministically partitioned data** set instead of a randomly partitioned one, as random partitioning will always cause positively biased results due to overlapping regions in patches.

The architecture of the proposed network is shown in Fig. 2. **It is mainly inspired by the philosophy of the well-known VGG Nets [7].** The SAR and optical image patches are passed through a stack of convolutional layers, where we make use of convolutional filters with a very small receptive field of  $3 \times 3$  rather than using larger ones, such as  $5 \times 5$  or  $7 \times 7$ . The reason is that  $3 \times 3$  convolutional filters are the smallest kernels to capture patterns in different directions, such as center, up/down, and left/right, but still have an advantage: **the use of small convolutional filters will increase the nonlinearities inside the network and thus make the network more discriminative.**

The convolution stride in our network is fixed to 1 pixel; the spatial padding of convolutional layer input is such that the spatial resolution is preserved after convolution, i.e., the padding is 1 pixel for the all  $3 \times 3$  convolutional layers in our network. Spatial pooling is achieved by carrying out seven max-pooling layers, which follow some of the convolutional layers. They are used to reduce the dimensionality of the feature maps. Max pooling is performed over  $2 \times 2$  pixel windows with stride 2.

The fusion stage of our proposed network is made up of two consecutive convolutional layers, followed by two fully connected layers. The convolutional layers consist of  $3 \times 3$  filters, which operate over the concatenated feature maps of the SAR and optical streams, in order to learn a fusion rule which minimizes the final loss function. Max pooling is omitted after the first convolutional layer in the fusion stage, and a stride of 2 is used in order to downsample the feature maps while preserving the spatial information [8]. The use of  $3 \times 3$  filters and the absence of max pooling after the first convolution allow the fusion layer to learn a fusion rule, which is somewhat invariant to spatial mismatches caused by the difference in imaging modalities. This is due to the fact that the fusion layer uses  $3 \times 3$  convolutions to learn relationships between the features while preserving nearby spatial information. The lack of max pooling means that these learned spatial relationships are preserved as not only the maximal response is considered, while the stride of 2 is used to reduce the feature size. The final stage of the fusion network consists of two fully connected layers: the first of which contains 512 channels; while the second, which performs one-hot binary classification, contains 2 channels.

In a nutshell, the convolutional layers in our network apart from the fusion layer generally consist of  $3 \times 3$  filters and follow two rules: 1) the layers with the same feature map size have the same number of filters and 2) the number of feature maps increases in the deeper layers, roughly doubling after each max-pooling layer (except for the last convolutional stack in each stream). All layers in the network are equipped with a rectified linear unit as an activation function, except the last fully connected layer, which is activated by a softmax function. Fig. 2 shows the schematic of the configuration of our network.

### B. Loss Function

We make use of the **binary cross-entropy loss for training our network**. Let  $X = (x_1^{\text{sar}}, x_1^{\text{opt}}), (x_2^{\text{sar}}, x_2^{\text{opt}}), \dots, (x_n^{\text{sar}}, x_n^{\text{opt}})$  be a set of SAR-optical patch pairs, where  $x_i^{\text{sar}}, x_i^{\text{opt}} \in \mathbb{R}^{D \times D}, \forall i = 1, \dots, n$  and  $\mathbf{y}_i$  is the one-hot label for the pair  $(x_i^{\text{sar}}, x_i^{\text{opt}})$  (with  $[1, 0]$  denoting a dissimilar pair, and  $[0, 1]$  denoting a similar pair). We then seek to minimize the binary

cross-entropy loss

$$E = \frac{1}{n} \sum_{i=1}^n y_i \cdot \log f(x_i^{\text{sar}}, x_i^{\text{opt}}, \theta) \quad (1)$$

where  $f(x_i^{\text{sar}}, x_i^{\text{opt}}, \theta)$  denotes the output vector of the softmax function when comparing the input pair  $(x_i^{\text{sar}}, x_i^{\text{opt}})$  with the current network parameters  $\theta$ .

### C. Configuration Details

Fig. 2 shows the full configuration of our network. Apart from the previously discussed architecture, we also make use of batch normalization after the activation function of each convolutional layer. This leads to an increase in the training speed and reduces the effects of internal covariate shift. In order to reduce overfitting during training, we made use of  $L_2$ -regularization, with  $\lambda = 0.001$ , for the convolution kernels of the SAR and optical streams, and dropout with a rate of 0.7 for the first fully connected layer.

## III. AUTOMATIC PATCH POOL GENERATION

For training and testing purposes, a large pool of corresponding and noncorresponding SAR and optical image patches is needed. While the classical work on deep matching for optical imagery can usually rely on easy-to-achieve optical patch pools (see, for example, the *Phototourism Patch data set* [4], [9]), annotating corresponding patches in VHR optical and SAR imagery of complex urban scenes is a highly nontrivial task even for experienced human experts. Thus, one of the contributions of this letter is the introduction of a fully automatic procedure for SAR-optical patch pool generation.

### A. “SARptical” Framework

In order to solve the challenge of automatic data set generation, we resort to the so-called “SARptical” framework of Wang *et al.* [10], an object-space-based matching procedure developed for mapping textures from optical images onto 3-D point clouds derived from SAR tomography. The core of this algorithm is to match the SAR and optical images in 3-D space in order to deal with the inevitable differences caused by different geometrical distortions. Usually, this would require an accurate digital surface model of the area to link homologue image parts via a known object space. In contrast, the approach in [10] creates two separate 3-D point clouds, one from SAR tomography and one from optical stereo matching, which are then registered in 3-D space to form a joint (“SARptical”) point cloud, which serves as the necessary representation of the object space. The flowchart of the approach can be seen in Fig. 3. In order to estimate the 3-D positions of the individual pixels in the images, the algorithm requires an interferometric stack of SAR images as well as at least a pair of optical stereoisimages. The matching of the two point clouds in 3-D guarantees the matching of the SAR and the optical images. Finally, we can project the SAR image into the geometry of the optical image via the “SARptical” point cloud and vice versa.

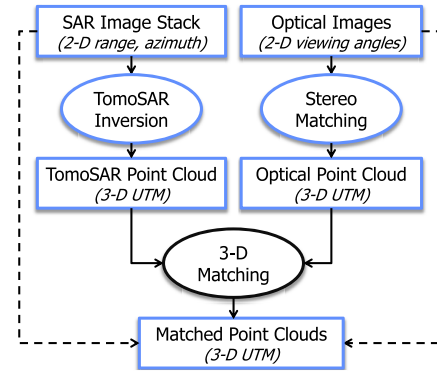


Fig. 3. Flowchart of the patch-pool generation procedure.

### B. Data Preparation

For the work presented in this letter, we made use of a stack of 109 TerraSAR-X high-resolution spotlight images of the city of Berlin, acquired between 2009 and 2013 with about 1-m resolution, and of nine UltraCAM optical images of the same area with 20-cm ground spacing. After the reconstruction of the “SARptical” 3-D point cloud, 8840 pixels with high SNR ( $>5$  dB) were uniformly sampled from the nonlocally filtered master SAR amplitude image and projected into the individual optical images, yielding a total of 10 108 corresponding optical pixels. The reason for the difference in pixel numbers is that each of the nine optical multiview stereoisimages is acquired from a different viewing angle, making it possible for each SAR image pixel to have up to nine corresponding optical image pixels. The actual number of corresponding optical pixels is dependent on the visibility of the SAR pixel from the respective optical point of view.

All SAR patches are centered at their corresponding SAR image pixels. Their size is fixed at  $112 \times 112$  pixels with a pixel spacing of about 1 m. In analogy, the optical patches are centered at the corresponding optical pixels. After resampling to adjust the pixel spacing, the SAR patches were rotated, so that both patches align with each other as a first approximation.

In order to reduce bias when training our network, we randomly selected a single correct optical correspondence for each SAR image patch during the final data set preparation. In addition, we randomly assign one wrong optical correspondence to each SAR patch in order to create negative examples. Thus, eventually, we end up with 17 680 SAR-optical patch pairs (see Fig. 1 for an example of the class of correct matches).

As final preprocessing steps, the optical patches were converted to gray scale, and all patches were normalized [11] to a radiometric range of  $[0; 1]$  with subsequent subtraction of their means.

### C. Patch Pool Partitioning

In order to provide a fair experimental design, we partition the patch pool in the following manner: 9724 (55%) of the patch pairs are used as training data set, 2652 (15%) as validation set, and 5304 (30%) as test data set. It has to be



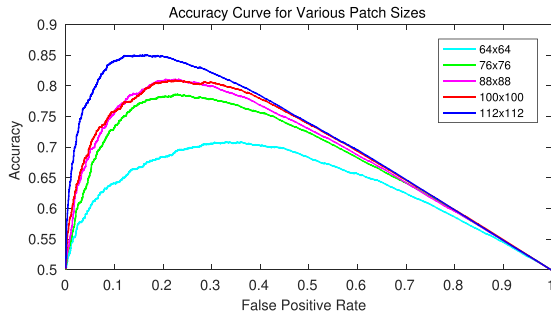


Fig. 4. Comparison of different patch sizes.

noted that we do not partition the patch pool on a purely randomized basis but rather resort to a deterministic partitioning method in order to avoid positively biased test results. The full extent SAR and optical images are first deterministically partitioned and then each partition is processed to generate positive and negative samples for training, validation, and testing, respectively.

#### IV. EXPERIMENTS AND RESULTS

##### A. Training Details

The network was trained using the Adam [12] optimization algorithm as it is computationally efficient and exhibits faster convergence than standard stochastic gradient descent methods. The optimization hyperparameters are fixed to  $\beta_1 = 0.9$  and  $\beta_2 = 0.999$  with a learning rate of  $\alpha_t = 0.0009$ . The learning rate was found via a grid search method on our training and validation data, while the  $\beta$ -parameters were kept at their recommended values. Prior to training the network, weight vectors were initialized using the truncated uniform distribution described in [13], and the bias vectors were initialized with zero values. Training was conducted with 2 Nvidia TitanX GPUs using class balanced, minibatches of 64 SAR-optical patch pairs (32 corresponding and 32 noncorresponding pairs) over 30 epochs; training took on average 25 min with a single forward pass taking around 3 ms to complete.

We trained five versions of our proposed network, each at a different patch size, in order to evaluate the effect of patch size on classification accuracy. Patch cropping was done on-the-fly with the new patch being cropped from the center of a larger patch—this was done as the center pixel is the point of correspondence between the SAR and optical patch. Furthermore, we seeded our random number generator with a fixed value of 0, at the start of training for each patch size, in order to prevent the randomization effects between networks.

##### B. Evaluation Results

We evaluate the proposed network with different input patch sizes using our testing patch pool (described in Section III), which has further been cropped around the center pixel to produce new testing pools with different patch sizes.

The accuracy versus false positive rate curves corresponding to different patch sizes can be seen in Fig. 4. Table I reports

TABLE I  
CONFUSION MATRIX VALUES FOR DIFFERENT PATCH SIZES

	64	76	88	100	112
TP	46.6%	61.6%	66.0%	69.8%	82.2%
TN	86.2%	88.0%	88.2%	86.0%	89.8%

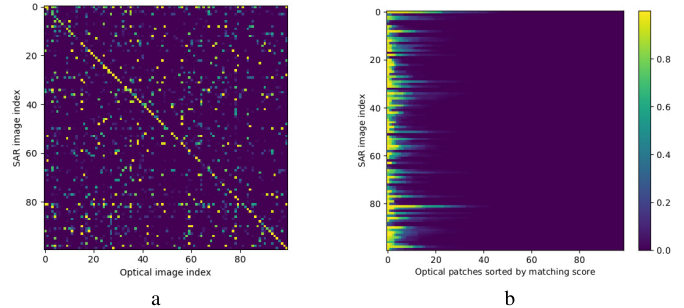


Fig. 5. Results of key-point matching experiment. (a) Confusion matrix showing the matching scores for all SAR and optical key-point patches. (b) Spread of incorrect matches ordered by the similarity score.

the corresponding confusion matrix values for our proposed network evaluated with each patch size; it is to be noted that the confusion matrix is the reflective of the network at the point of highest overall performance for each patch size.

##### C. Key-Point Matching Results

In order to evaluate the proposed network's performance in a real-world, key-point matching scenario, we selected 100 neighboring TomoSAR key-points in the SAR image and extracted the corresponding SAR and optical patch pairs. We selected these key points from a localized area within our test set so as to reproduce the conditions found in a real-world key-point matching application. We then compared every SAR and optical patch in the selected patch set in order to determine the performance of our proposed network in the presence of large numbers of potential mismatches.

In Fig. 5(a), we can see a matrix depicting the similarity scores of the various pair comparisons, where corresponding SAR and optical patches are given the same index number. It should be noted that in determining a binary value for correspondence, a threshold is applied to these similarity scores. Fig. 5(b) shows the sorted scores for all nonsimilar optical patches, making it easier to see the number and strength of incorrect matches in the patch pool.

#### V. DISCUSSION

Generally, the results summarized in Section IV-B indicate a promising discriminative power of the proposed network. However, the following major points must be considered when interpreting the results.

##### A. Influence of the Patch Size

As Table I and Fig. 4 clearly indicate, the patch size strongly affects the discriminative power of the network. This result is likely due to the effects of distortions in SAR images, which

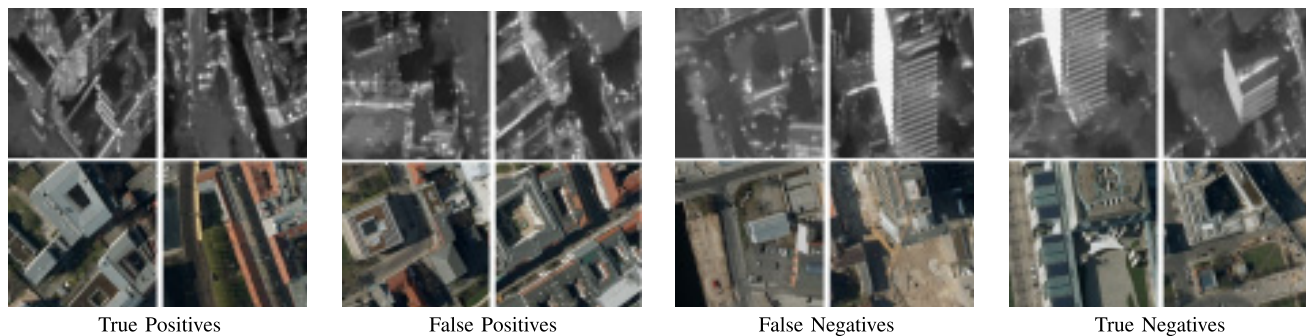


Fig. 6. Exemplary patch correspondence results.

are acquired in a range-based imaging geometry. Thus when patches are cropped to smaller regions, we likely crop out defining features, which are used for matching between the SAR and optical domain. This can be intuitively understood by referring to Fig. 1, where we can see the effects of layover and multipath reflections of the building in the SAR image and a near top down view of the same building in the optical image. Taking away explanatory context will thus render the matching more difficult. All further discussion will be with reference to the results we obtained using the largest patch size, 112 pixels.

#### B. Comments on the Discriminative Power of the Proposed Network

In summary, our approach obtains an accuracy exceeding 77% on a separate test data set when fixing the false positive rate to 5%, which falls into the same order of magnitude as what can be achieved using the powerful handcrafted HOPC descriptor in combination with an  $L_2$ -norm cost function [3].

Furthermore, our approach produced a clear diagonal pattern in Fig. 5(a), which depicts its ability to accurately determine the correct correspondence in a key-point matching scenario. Upon further investigation, it was found that the network achieved 43% top-1 matching accuracy and 74% top-3 accuracy, while 8% of points had no valid matches detected within the key-point set. This was found to be due to large amounts of layover and extreme differences in view point between the SAR and optical patches (see false negatives in Fig. 6).

#### C. Possible Reasons for False Predictions

From the randomly chosen prediction examples shown in Fig. 6, it can be observed that many of the false positives and false negatives are erroneously matched under extreme differences in viewing angle between the SAR and optical patches. While this may partially be solvable by resorting to larger patch sizes, providing valuable context, there might be a need to exclude image parts with all too strong distortions from further processing.

### VI. CONCLUSION

In this letter, a pseudo-siamese CNN for learning to identify corresponding patches in VHR SAR and optical images in a fully automatic manner has been presented. A first evaluation has shown promising potential with respect to multisensor

key-point matching procedures. In order to ensure transferability to other applications not based on key points, e.g., dense matching, we will work on the generation of additional training patches, whose center pixel does not rely on specific key points. In addition, we will test the approach on data coming from a completely different source. In the end, we expect this letter to help paving the way for generalized SAR-optical image matching procedures.

#### ACKNOWLEDGMENT

The authors would like to thank the NVIDIA Corporation for donating the Titan X Pascal GPU used in this letter.

#### REFERENCES

- [1] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 10, pp. 1615–1630, Oct. 2005.
- [2] M. Schmitt and X. X. Zhu, "Data fusion and remote sensing: An ever-growing relationship," *IEEE Geosci. Remote Sens. Mag.*, vol. 4, no. 4, pp. 6–23, Dec. 2016.
- [3] Y. Ye, J. Shan, L. Bruzzone, and L. Shen, "Robust registration of multimodal remote sensing images based on structural similarity," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 5, pp. 2941–2958, Mar. 2017.
- [4] X. Han, T. Leung, Y. Jia, R. Sukthankar, and A. C. Berg, "MatchNet: Unifying feature and metric learning for patch-based matching," in *Proc. CVPR*, Boston, MA, USA, Jun. 2015, pp. 3279–3286.
- [5] S. Zagoruyko and N. Komodakis, "Learning to compare image patches via convolutional neural networks," in *Proc. CVPR*, Boston, MA, USA, 2015, pp. 4353–4361.
- [6] L. Mou, M. Schmitt, Y. Wang, and X. X. Zhu, "A CNN for the identification of corresponding patches in SAR and optical imagery of urban scenes," in *Proc. JURSE*, Dubai, United Arab Emirates, Mar. 2017, pp. 1–4.
- [7] K. Simonyan and A. Zisserman. (2014). "Very deep convolutional networks for large-scale image recognition." [Online]. Available: <https://arxiv.org/abs/1409.1556>
- [8] A. Handa, M. Bloesch, V. Pătrăucean, S. Stent, J. McCormac, and A. Davison, "gvnn: Neural network library for geometric computer vision," in *Proc. ECCV Workshops*, Amsterdam, The Netherlands, 2016, pp. 67–82.
- [9] S. A. J. Winder and M. Brown, "Learning local image descriptors," in *Proc. CVPR*, Minneapolis, MN, USA, Jun. 2007, pp. 1–8.
- [10] Y. Wang, X. Zhu, B. Zeisl, and M. Pollefeys, "Fusing meter-resolution 4-D InSAR point clouds and optical images for semantic urban infrastructure monitoring," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 1, pp. 14–26, Jan. 2017.
- [11] Q. Wang, C. Zou, Y. Yuan, H. Lu, and P. Yan, "Image registration by normalized mapping," *Neurocomputing*, vol. 101, pp. 181–189, Feb. 2013.
- [12] D. P. Kingma and J. Ba. (2014). "Adam: A method for stochastic optimization." [Online]. Available: <https://arxiv.org/abs/1412.6980>
- [13] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. AISTATS*, Sardinia, Italy, 2010, pp. 249–256.