Learning geometric soft constraints for multi-view instance matching across street-level panoramas

Ahmed Samy Nassar, Nico Lang, Sébastien Lefèvre, Jan Wegner

Learning geometric soft constraints for multi-view instance matching across street-level panoramas

Ahmed Samy Nassar[1,2], Nico Lang[2], Sébastien Lefèvre[1], Jan D. Wegner[2]
[1]*IRISA, Université Bretagne Sud*
[2]*EcoVision, ETH Zurich*
{ahmed-samy-mohamed.nassar, sebastien.lefevre}@irisa.fr, {nico.lang, jan.wegner}@geod.baug.ethz.ch

***Abstract*—We present a new approach for matching tree instances across multiple street-view panorama images for the ultimate goal of city-scale street-tree mapping with high positioning accuracy. What makes this task challenging is the strong change in view-point, different lighting conditions, high similarity of neighboring trees, and variability in scale. We propose to turn (tree) instance matching into a learning task, where image-appearance and geometric relationships between views fruitfully interact. Our approach constructs a siamese convolutional neural network that learns to match two views of the same tree given many candidate tree image cut-outs and geographic information of the two panorama images. In addition to image features, we propose utilizing location information about the camera and the tree. Our method is compared to existing patch matching methods to prove its edge over state-of-the-art. This takes us one step closer to the ultimate goal of city-wide tree mapping based solely on panorama imagery to benefit city administration.**

I. Introduction

Monitoring street-side objects in public spaces in cities is a labor-intensive and costly process. One strategy that can complement greedy city surveillance and maintenance efforts by field crews as done in practice today is crowdsourcing information through geo-located images. In previous work [1]–[3], we have come up with an automated pipeline that catalogues trees from street-view panorama images by detecting, localizing and classifying them into species. A bottleneck of the existing method is its low geo-positioning accuracy of detected trees, which is caused by (i) often relying on only one detection for estimating the geo-location and (ii) combining detections of different trees into a single position if reasoning across multiple views. In order to reduce false matches and to improve geo-positioning of trees, we propose to explicitly exploit image evidence as well as soft geometric constraints to match images of the same tree across multiple panoramas. We formulate this problem as an instance matching task, where the typical warping function between multiple views of the same tree (Fig. 1) in street-view panoramas is learned together with the geometric configuration. More precisely, instead of merely relying on image appearance for instance matching, we insert heading, geo-location and further geometric parameters of the different views to the learning process. The CNN learns to correlate typical geometric configurations with corresponding warping functions to disentangle multiple matching candidates in case of ambiguous image evidence. This spatial information can help boosting instance matching scores, which resolves

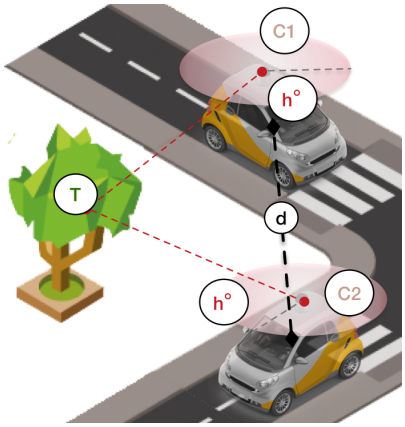

Figure 1: C*: Camera with geo-position. T: The tree has its actual geographic coordinates, and location within the panorama. $h°$: heading angle inside panorama. d: Distance between cameras.

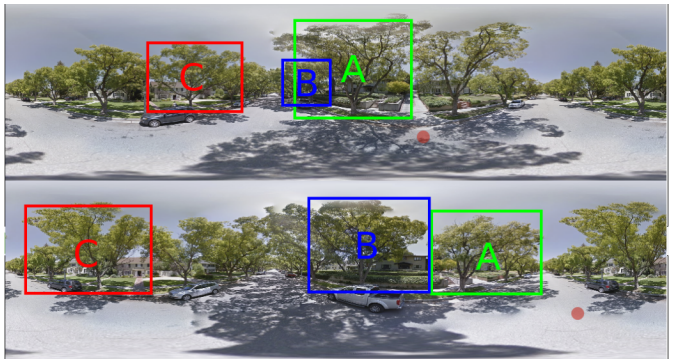

Figure 2: Tree instance matching problem (color indicates matches): each tree is photographed from multiple different views, changing its size, perspective, and background. Note that many trees look alike.

difficult situations where many similar trees exist in close proximity as presented in Figure 2.

In this work, we propose to learn image matching with soft geometric constraints to improve geo-positioning of street trees from panorama images. Our method builds upon the siamese architecture proposed originally by [4]. The main concept of siamese CNNs is constructing two identical network branches that share (at least partially) their weights. Features are computed for both input images and then compared to estimate the degree of similarity. This can be achieved by

either evaluating a distance metric in feature space or by evaluating the final classification loss. We build a siamese CNN to match images of the same tree across multiple street-view panoramas. Google street-view provides access to a huge amount of street-level images that can be used to construct very large data sets for deep learning approaches. Here, we use it to build a multiview data set of street-trees, which is used as a testbed to learn instance matching with soft geometric constraints based on a siamese CNN model.

There is a great number of research efforts that try to match objects across multiple views. Some traditional methods [5] use SIFT [6] to extract features, and match them. The most similar problem to our task seems the person re-id problem, which tries to identify a person in multiple views. Several methods [7] similarly employ siamese CNNs to solve the problem. However, our problem is different in that objects are static, but appear from very different viewing angles and distances in contrast to the face identification [8] problem.

Our main contribution is a modified siamese CNN that jointly learns geometric constellations of panorama acquisitions with the appearance information in the images. This will further on help us in our main pipeline to better geo-position trees, and to classify species, stress level, etc.

II. Instance Matching with Soft Geometric Constraints

An overview of the proposed model architecture is shown in Fig. 3. The main idea is that corresponding images of the same tree should follow the basic principles of stereo- (or multiview) photogrammetry if the relative orientation between two or more camera viewpoints can be established. Directly imposing hard constraints based on the rules of, for example, forward intersection is hard. An unfavorable base-to-height ratio, i.e. trees on the street-side get very close to the camera but the distance between two panorama acquisitions is significantly larger, makes dense matching impossible. The perspective of the object changes too much to successfully match corresponding image pixels. Moreover, the heading and geolocation (that are recorded in the metadata of street-view panoramas) are often inaccurate due to telemetry interference or other causes. We thus propose to implicitly learn the distribution of geometric parameters that describe multi-view photogrammetry together with the image appearance of the objects. Our assumption is that this approach will enable cross-talking between image evidence and geometry. For example, if a same tree appears with the same size in two images (but very different perspective), the triangle that connects both camera positions and the tree must be roughly isosceles. That is, the tree is located in the middle between both camera standpoints. Conversely, a tree that is viewed from the same perspective (very similar image appearance) but appears rather small will point at a pointy triangle with one very long leg (longer than the baseline) and another shorter leg. More literally speaking, the tree will most likely be situated outside the baseline between the two cameras.

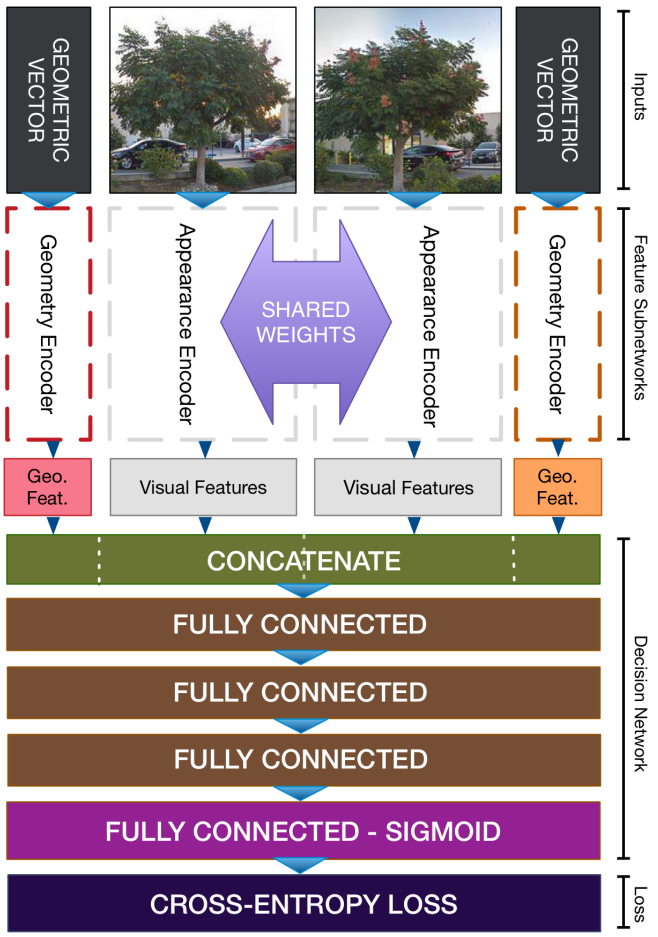

Figure 3: Diagram showing the overall network architecture.

We use geometric features composed of $\{[C^*_{lat}, C^*_{lng}, d, h^\circ]\}$, where C represents the panorama image geolocation, d denotes the distance between the cameras in meters, and h° is the heading angle of the tree inside the panorama image. We add these geometric cues to image evidence similar to [9] who also merge multi-modal data inside a single CNN architecture to minimize a joint loss function. Our modified siamese CNN processes image crops, and geometric features jointly. Two image patches are resized to 224x244 pixels and fed to the two network streams separately. In addition, a vector with the four geometric features is fed to the network as shown in Fig. 3.

After trying various CNN architectures as presented in Sec. III, we found MatchNet [10] to perform best in our scenario (Fig. 3). The network is composed of four streams that extract image and geometry features with the layers shown in "Feature Subnetworks". We further modified the original MatchNet architecture by adding batch normalization [11] and dropout layers. As shown in Fig. 3, each block contains batch normalization, a Rectified Linear Unit (ReLU) [12], pooling, and a convolutional layer. Features from the four separate streams are concatenated and passed to the decision-making part of the network, which computes the similarity. This network part is composed of four FC (fully connected)

networks. The final FC layer uses a sigmoid activation function to limit the output space in the range [0,1]. We use a binary cross entropy loss to directly classify whether an image pair shows the same tree or not.

Our Siamese CNN shares weights of the appearance encoders, as suggested by [13] when dealing with the same modality. Thus, both of our appearance encoder subnetworks are identical and with shared network parameters. An important insight is that sharing network parameters of the geometry encoder subnetworks is not successful and we thus keep two separate streams. The major reason is that our model learns a warping function between two images while implicitly keeping track of the relative orientation of both images. Mingling both camera orientations is counterproductive and looses the main information gain through the introduction of geometry.

III. Experiments

Our implementation is based on Keras [14]. Weights of the network are initialized using the "Glorot uniform initializer" [15], the initial learning rate is set to 0.0001 with ADAM [16] as the optimizer, and the dropout rate is set to 0.3.

A. Dataset

We test our approach on a new dataset of Pasadena, California, USA, which extends the existing urban trees dataset of our previous work [2], [3]. It is generated from an existing KML-file that contains rich information (geographic position, species, trunk diameter) of 80,000 trees in the city of Pasadena. For every tree we downloaded the closest 4 panoramic images of size 1664x832 pixels from Google Street View. A subset of 4400 trees with four views each is chosen, leading to 17,600 images in total plus meta-data. Note that the Pasadena inventory contains only street-trees, which makes up roughly 20% of all city trees. We draw bounding boxes around all street-trees per panorama image, which results in 47,000 bounding boxes in total. A crucial part of the labeling task is to label corresponding images of the same tree in the 4 closest views as shown in Fig. 1. Our final dataset is composed of panoramic images containing labeled trees (and matches between four tree images per tree), the panorama meta-data (geographic location and heading of the camera), and the geo-position per tree. Note that the geo-position per tree is used during training to establish ground truth parameters of our geometric features. It is not used during testing, but geometric parameters are directly derived from the individual panoramas.

B. Evaluation strategy

We split the dataset into 3 subsets with 70% for training, 15% for validation, and 15% for testing paying attention to no overlap between train and test data panoramas. Each tree comes with four image patches from different views. Each image patch comes with a feature vector that contains geometric cues as described in Sec. II. For training the positive match category, we insert matching image patch pairs from the same tree with the geometry feature vectors to our model. Negative pairs of the rejection category are generated by randomly picking two image patches from two different trees. Initial tests showed that most mismatches occur at neighboring trees because geometry is least discriminative in such cases, the warping function is very similar, and the often same species leads to very similar appearance as well as a common background. We therefore add many negative example pairs from neighboring trees to make the classifier more robust.

	Method	Network	mAP
Appearance Only	*SimpleStacked*	MatchNet	0.762
	SimpleStacked	ResNet-50	0.805
	Siamese	FaceNet Modified	0.808
	Siamese	ResNet-50	0.828
	Siamese	MatchNet	0.843
W/ Geometry	*Ours*	FaceNet Modified	0.842
	Ours	ResNet-50	0.863
	Ours	MatchNet	**0.871**

Table I: Matching results of our method with different base network architectures (*Ours*) compared to baseline methods that use only appearance information and no geometric cues.

C. Results

We compare our approach against several baselines like a standard Siamese CNN and other image patch matching works [17] to benchmark its performance:

- *SimpleStacked*: We stack both images and load each pair as a six-layer image as suggested in [17]. MatchNet and ResNet50 are used as architectures for classification into matches and non-matches.
- *Siamese*: A standard siamese CNN composed of two identical subnetworks (MatchNet, FaceNet, ResNet-50) with a decision network part that decides whether the image pair matches or not.
- *Ours*: We augment the architectures of *Siamese* with geometry as described in Sec. II.

Results shown in Tab. I indicate that *Ours* consistently outperforms all baseline methods regardless of the base network architecture. Any architecture with added geometric cues does improve performance. This finding suggests that adding geometric features helps reducing matching errors in general. Learning soft geometric constraints of typical scene configurations helps differentiating correct from wrong matches in intricate situations. Overall, *Ours* with the MatchNet architecture performs best.

Examples for correct classifications as not matching and matching for hard cases are shown in Figs. 4 and 5, respectively. *Ours* with the MatchNet architecture is able to correctly classify pairs of similar looking trees in the same proximity as not matching (Fig. 4), which was the main goal of this work. It also helps establishing matches correctly in difficult situations of very different viewing angles and occlusion (Fig. 5).

Figure 4: Top & bottom: Images of neighboring trees of very similar appearance and background that are correctly classified as not matching with our method (*Ours* with MatchNet).

Figure 5: Top & bottom: Correct matching of images of the same tree in difficult situations (*Ours* with MatchNet). The target tree appears in two panoramas captured from very different angles, partial occlusion (only top) and different scene illuminations.

IV. Conclusion

We have presented a modified siamese CNN architecture that jointly learns distributions of appearance-based warping functions and geometric scene cues for (tree) instance matching in the wild. Instead of sequentially imposing hard thresholds based on multiview photogrammetric rules, joint learning of appearance and geometry enables cross-talking of evidence inside a single network. While our network design is a slightly adapted version of standard siamese CNNs and exploits existing architectures like MatchNet, it already shows promising performance. Better tree instance matching across multiple different views helps establishing object correspondence, to ultimately improve geo-localization of trees in the bigger framework. Our hope is that this idea of "learning photogrammetry" and combining it with object appearance will unleash a whole new line of research. For example, learned, soft photogrammetric constraints can also help improving object detection across multiple views. Learning photogrammetric constraints as soft priors jointly with image evidence will help in many situations where camera and object poses are ill-defined, noisy, or partially absent.

References

[1] J. D. Wegner, S. Branson, D. Hall, K. Schindler, and P. Perona, "Cataloging public objects using aerial and street-level images-urban trees," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 6014–6023.

[2] S. Branson, J. D. Wegner, D. Hall, N. Lang, K. Schindler, and P. Perona, "From google maps to a fine-grained catalog of street trees," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 135, pp. 13–30, 2018.

[3] S. Lefèvre, D. Tuia, J. D. Wegner, T. Produit, and A. S. Nassaar, "Toward seamless multiview scene analysis from satellite to street level," *Proceedings of the IEEE*, vol. 105, no. 10, pp. 1884–1899, 2017.

[4] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a "siamese" time delay neural network," in *Advances in Neural Information Processing Systems*, 1994, pp. 737–744.

[5] J. Joglekar, S. S. Gedam, and B. K. Mohan, "Image matching using sift features and relaxation labeling technique—a constraint initializing method for dense stereo matching," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 9, pp. 5643–5652, 2014.

[6] D. G. Lowe, "Object recognition from local scale-invariant features," in *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, vol. 2. Ieee, 1999, pp. 1150–1157.

[7] W. Li, R. Zhao, T. Xiao, and X. Wang, "Deepreid: Deep filter pairing neural network for person re-identification," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 152–159.

[8] M. Aly, "Face recognition using sift features," *CNS/Bi/EE report*, vol. 186, 2006.

[9] E. Park, X. Han, T. L. Berg, and A. C. Berg, "Combining multiple sources of knowledge in deep cnns for action recognition," in *IEEE Winter Conference on Applications of Computer Vision*, 2016, pp. 1–8.

[10] X. Han, T. Leung, Y. Jia, R. Sukthankar, and A. C. Berg, "Matchnet: Unifying feature and metric learning for patch-based matching," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3279–3286.

[11] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[12] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *International Conference on Machine Learning*, 2010, pp. 807–814.

[13] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1701–1708.

[14] F. Chollet *et al.*, "Keras," https://keras.io, 2015.

[15] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *International Conference on Artificial Intelligence and Statistics*, 2010, pp. 249–256.

[16] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[17] S. Zagoruyko and N. Komodakis, "Learning to compare image patches via convolutional neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 4353–4361.