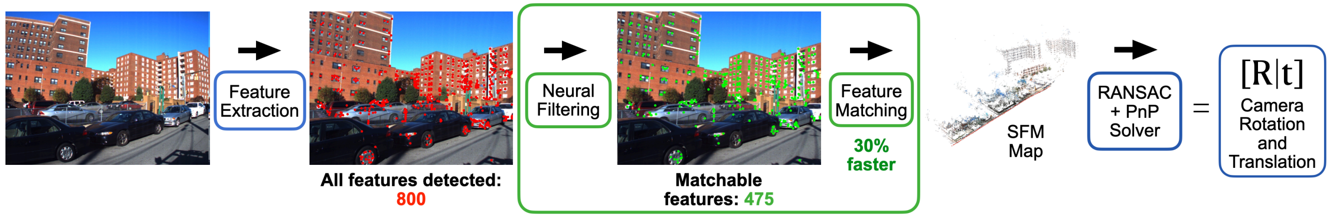
BMVC 2023

CYENS CENTRE OF EXCELLENCE   UNIVERSITY OF BATH   EPFL

Neural Feature Filtering for Faster Structure-from-Motion Localisation

Alexandros Rotsidis[1,2], Wang Yuxin[3], Yiorgos Chrysanthou[2], Christian Richardt[1]

ar2056@bath.ac.uk, yuxin.wang@epfl.ch, y.chrysanthou@cyens.org.cy, christian@richardt.name



Introduction

In this work address the crucial task of camera localization in offline maps, essential for applications like augmented reality, self-driving cars, and robotics. Camera localization pipelines typically involve feature detection, matching, outlier filtering, and pose estimation, with feature matching being a common bottleneck. The research focuses on removing dynamic points that can be outliers and slow down feature matching.

Proposed Method

The approach involves training a scene-specific neural network to assess the reliability of each detected feature descriptor (matchability). By doing so, the method can efficiently select the most matchable keypoints for subsequent pose estimation steps. Importantly, this method is adaptable to any existing structure-from-motion data. The study evaluates its performance on extensive indoor and outdoor datasets and compares it to two similar methods addressing the same issue. Additionally, the code for the proposed method is made available for use.

Neural Network Details

The inputs to the neural network are the SIFT descriptor, x and y pixel location, pixel RGB value, dominant orientations, size, response, octave, and orientation, totalling 138 features for the input vectors, one vector for each image keypoint. The neural network has six layers consisting of an input layer of 138 nodes, three more layers of 276 nodes, one layer of 138 nodes and the last output layer of one node. Cost function for imbalanced data from Wang et al.[1]

$$MSFE = \frac{1}{2}((FPE + FNE)^2 + (FPE - FNE)^2),$$

$$FPE = \frac{1}{N}\sum_{i=1}^{N}\sum_{n}\frac{1}{2}(d_n^{(i)} - y_n^{(i)})^2 \quad \text{and}$$

$$FNE = \frac{1}{P}\sum_{i=1}^{P}\sum_{n}\frac{1}{2}(d_n^{(i)} - y_n^{(i)})^2.$$

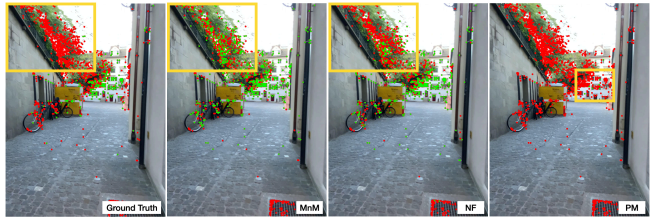
Results

We compare our method (*NF*), to methods from Papadaki and Hansch[2], and Hartmann et al.[3], (*MnM* and *PM*)

Dataset	CMU			LaMAR			Retail shop		
Method	MnM	NF	PM	MnM	NF	PM	MnM	NF	PM
T. Er. [m/cm]	0.59	1.30	12.13	2.58	2.65	19.27	0.52	0.54	1.52
Rot. Er. [°]	0.38	0.34	10.23	2.20	2.44	41.73	0.29	0.30	0.69
Feat. Red. [%]	37.64	70.73	93.06	49.91	66.92	99.33	26.39	49.92	91.58
F.M. Time (ms)	104	66	19	2,254	1,703	45	1,020	754	142
mAA [%]	98.99	98.54	86.51	96.21	95.36	41.87	97.91	97.94	82.74

Please refer to paper for more elaborate results.

Mean metrics for each dataset. For CMU and LaMaR, we report the translation error in metres (m), and for the retail shop in centimetres (cm). Our neural filtering method (NF) returns that highest feature keypoints reductions, which leads to faster feature matching, and also does not deteriorate the pose errors, unlike PM which returns high reductions but also high errors.



A random frame from LaMAR LIN, and the predictions non-matchable (red), and matchable (green). For the ground truth, the green points are the keypoints that have a 3D point matched in the live map, red if not. An area of interest is highlighted in yellow. MnM fails to detect the leaves in the top left corner as non-matchable compared to NF, which discards more points on leaves. PM fails to perform adequately and returns only a small number of points on the building.

Training Data, CMU LaMAR and Retail Shop datasets

A point cloud generated by SfM holds enough information that can be used to train a classifier to predict if a point is matchable or not. Each keypoint is matched to a 3D point, or not if that keypoint was not triangulated. For example, a keypoint detector may detect 500 keypoints in an image, but only 100 or so may be triangulated to form 3D points This leads to imbalanced data.

Dataset	Positive (%)	Negative (%)
CMU	30	70
LaMAR	14	86
Retail shop	60	40

Conclusion and Future Work

In this work, we proposed and evaluated a single neural network architecture to speed up feature matching by discarding superfluous features before they are matched to an existing map. We showed that taking into consideration the imbalanced data nature of the problem, we can achieve a better balance between feature matching speed and pose estimation errors compared to existing methods. Our results show that neural networks when adapted for the imbalanced data, are a promising option for efficiently filtering out unmatchable image descriptors, which can significantly reduce downstream computation time. We believe that more dynamic keypoints can be discarded while keeping the pose errors minimal if additional metadata can be added to the training data, e.g. semantic information.

[1] Shoujin Wang, Wei Liu, Jia Wu, Longbing Cao, Qinxue Meng, and Paul J Kennedy. Training deep neural networks on imbalanced data sets. In International Joint Conference on Neural Networks, pages 4368–4374, 2016.
[2] Alexandra I. Papadaki and Ronny Hansch. Match or no match: Keypoint filtering based on matching probability. In CVPR Workshops, pages 1014–1015, 2020.
[3] Wilfried Hartmann, Michal Havlena, and Konrad Schindler. Predicting matchability. In CVPR, pages 9–16, 2014

1. University of Bath, UK
2. CYENS Centre of Excellence Nicosia, Cyprus
3. École polytechnique fédérale de Lausanne, Switzerland