Internship report: Toward Fast Deep Visual Place Recognition

Rémi Piau
Supervisor: **Dr. Daniel Cremers**
Advisors: *Nikolaus Demmel & Lukas Köstler & Nan Yang*

CVG TUM, ENS Rennes

August 2020

Table of contents

Visual Simultaneous Localization And Mapping

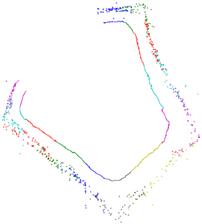
Goal

Compute the map of the surrounding environnement and the route taken inside this map at the same time.

VSLAM Drifts

Errors

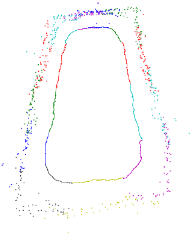
- Acquisition
- Computation
- Estimation



Source: Williams et al. [6]

Loop Closure to the Rescue!

Add map constrains when closing a loop to negate error.

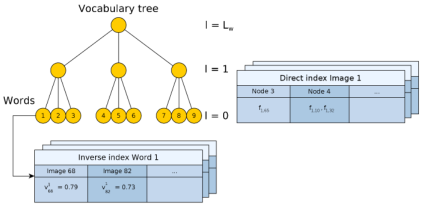


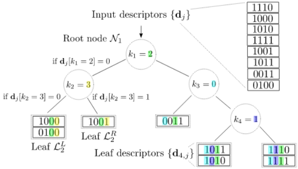
Loop detection cannot be based on the computed position.

Feature Based Visual Place Recognition

- DBOW
- HashBOW
- HBST

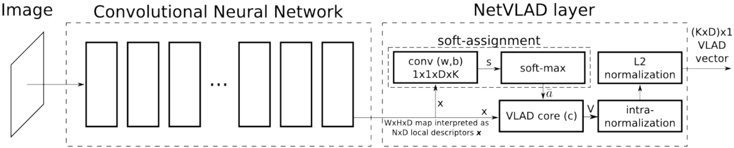
(Sources:
Gálvez-López&Tardós [2],
Schlegel&Grisett [4])
(Implementation: Tim Stricker [5])





Deep Learning Based Visual Place Recognition

Based on the Vector of Locally Aggregated Descriptors [3]



Source: Arandjelović et al. [1]

Why NetVLAD

- Quite popular
- Deep learning implies GPU based
- No benchmark on GPU & CPU

Current Implementations

- Original implementation proprietary matlab language
- Tensorflow 1 implementation without training
- Pytorch implementation with training procedure
- Keras implementation without training

Our Implementation

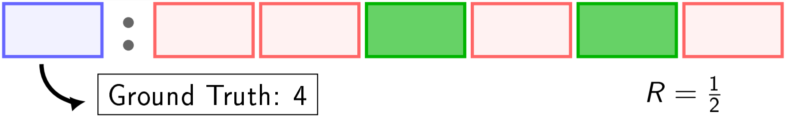
Key points

- Tensorflow 2 Keras implementation
- Training enabled
- Weight transfer from TF1 implementation & same exposed methods
- Checked against reference & TF1 implementations
- Quantization !

Metric

How well?

- Precision: number of queries with at least one correct result retrieved divided by the number of queries
- Recall: number of correct results divided by the number of ground truth results



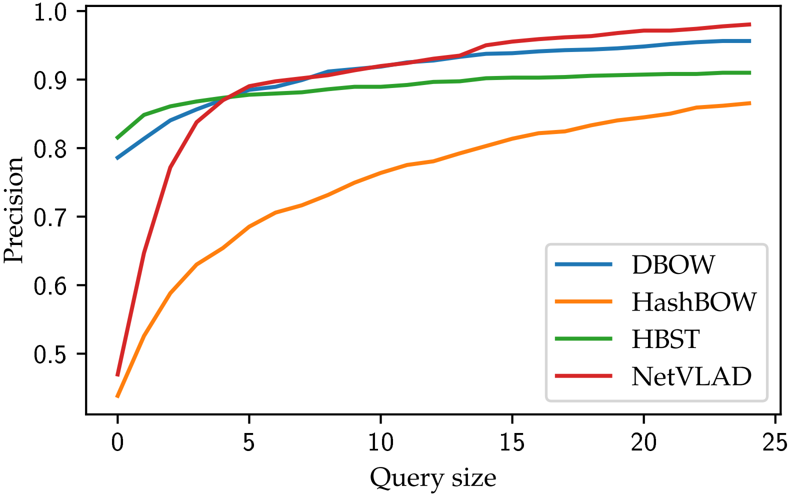
How fast?

Runtime on different phase of vpr recognition. Time in function of image size

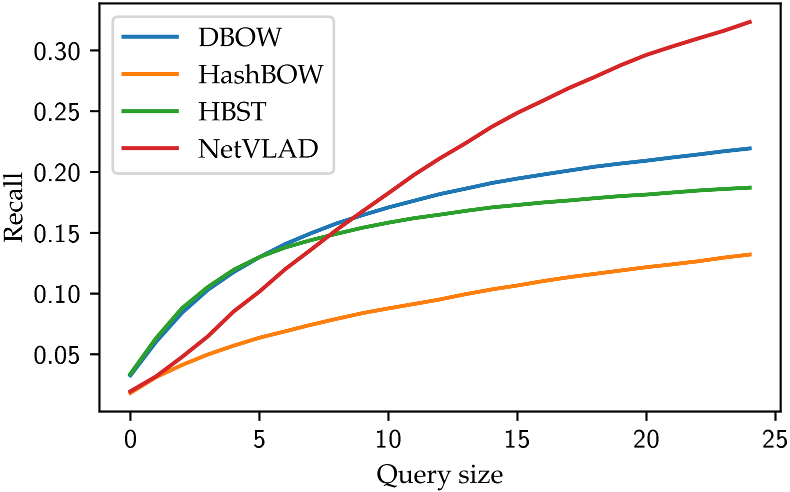
Quantitative comparison

(Source: Oxford FABMAP City Centre Dataset)

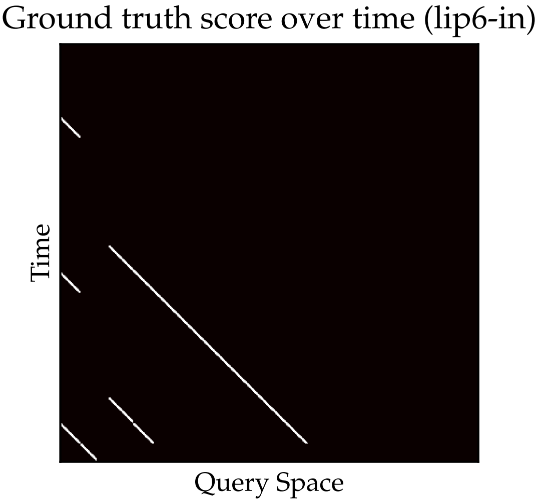
Precision


Precision
Query size
DBOW
HashBOW
HBST
NetVLAD
1.0
0.9
0.8
0.7
0.6
0.5
0
5
10
15
20
25

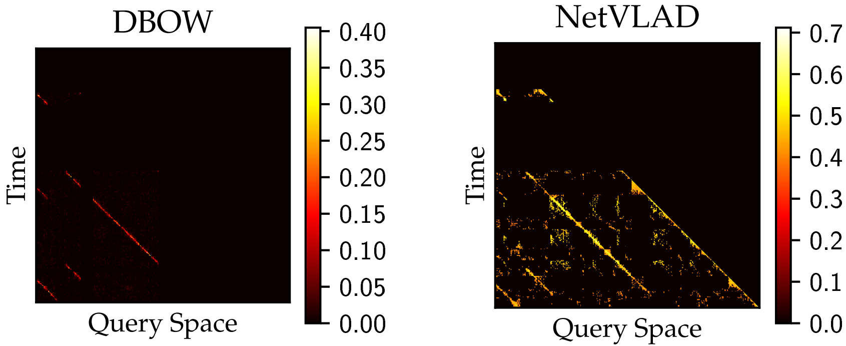
Recall


DBOW
HashBOW
HBST
NetVLAD
Recall
0.30
0.25
0.20
0.15
0.10
0.05
0
5
10
15
20
25
Query size

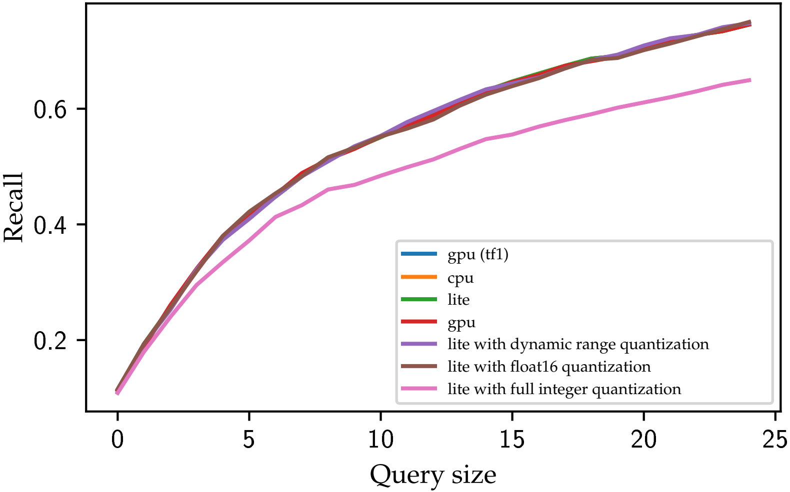
Another Story: Score Over Time



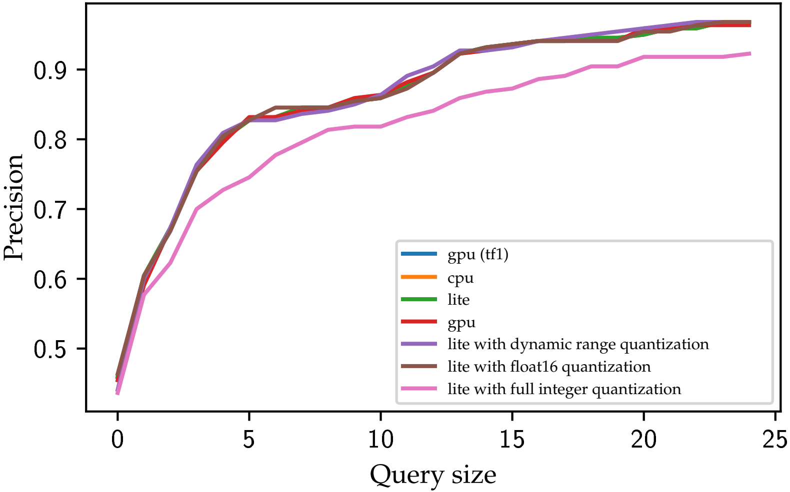
Another Story: Score Over Time



Quantization: Recall



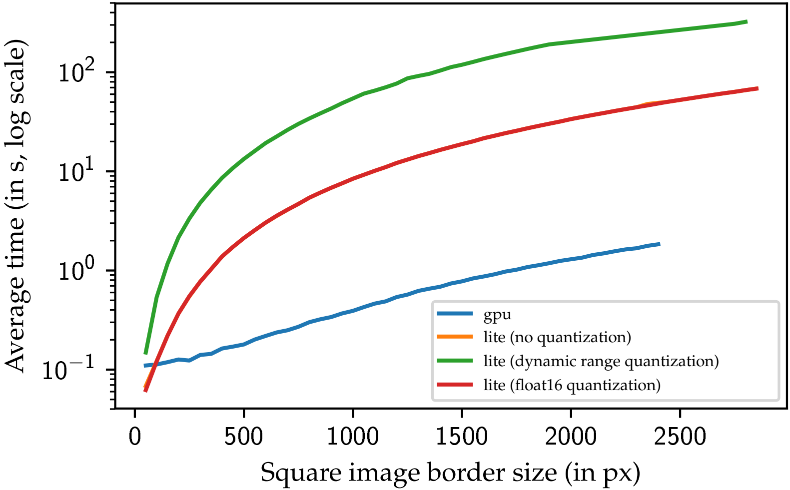
Quantization: Precision



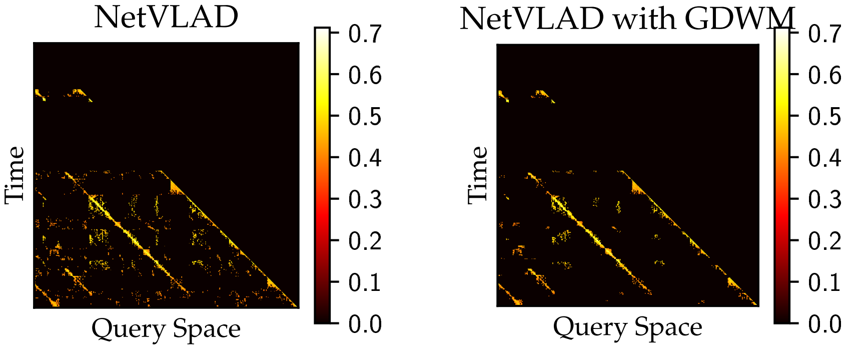
Quantization: Speed on Small Dataset

Method	Query Mean Time (s)	Slower by	
DBOW	0.00324	1x	
GPU	3.46	1068x	1x
lite with float16 quantization	6.89	2127x	1.99x
lite	7.21	2227x	2.08x
CPU	10.1	3119x	2.91x
lite with dynamic range quantization	38.2	11809x	11.1x
lite with full integer quantization	552	170725x	160x

Quantization: Speed vs Size



Greedy Double Windowed Mean



Conclusion

Take Over

- NetVLAD has better precision than classical methods
- Still a speed bottleneck for deep learning methods
- Tensorflow quantization make things slower on a desktop computer
- Post filtering is a simple way to improve results at low cost

Future Work

- Test further NetVLAD configurations
- Improve post filtering (delayed & inertia)

References

[1] Arandjelović, R., Gronat, P., Torii, A., Pajdla, T., and Sivic, J.
NetVLAD: CNN architecture for weakly supervised place recognition.
In *IEEE Conference on Computer Vision and Pattern Recognition* (2016).

[2] Galvez-López, D., and Tardos, J. D.
Bags of binary words for fast place recognition in image sequences.
IEEE Transactions on Robotics 28, 5 (2012), 1188–1197.

[3] Jégou, H., Douze, M., Schmid, C., and Pérez, P.
Aggregating local descriptors into a compact image representation.
In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2010), pp. 3304–3311.

[4] Schlegel, D., and Grisetti, G.
Hbst: A hamming distance embedding binary search tree for feature-based visual place recognition.
IEEE Robotics and Automation Letters 3, 4 (2018), 3741–3748.

[5] Stricker, T.
Efficient techniques for accurate visualplace recognition.
`https://vision.in.tum.de/_media/members/demmeln/stricker2020ma.pdf`.
Accessed: 2020-08-10.

[6] Williams, B., Cummins, M., Neira, J., Newman, P., Reid, I., and Tardos, J.
An image-to-map loop closing method for monocular slam.
In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems* (2008), pp. 2053–2059.

Greedy Double Windowed Mean

Parameters: first window size K, second window size K', threshold T.

For a query result vector $(r_{n,i})_{i \in 1..S}$ of size S at step n. and the memory of K previous results $(r_{n-j,i})_{i \in 1..S}$, $j \in 1..K$. We first define the mean vector as:

$$m_{n,i} = \frac{1}{K+1} \sum_{j=1}^{K+1} r_{n-j,i}$$

We define the mask at step n for all $i \in 1..S$ as:

$$M_{n,i} = \begin{cases} r_{n,i}, & \text{if } \displaystyle\sum_{j=\max(-K'/2,1)}^{\min(K'/2,S)} m_{n,j} \geq T \sum_{j=0}^{S} m_{n,j} \\ 0, & \text{otherwise} \end{cases}$$