Boosting the Performance of Scene Recognition via Offline Feature-Shifts and Search Window Weights

Chu-Tak Li, Wan-Chi Siu, Life-FIEEE and Daniel P.K. Lun, SrMIEEE
Centre for Multimedia Signal Processing, Department of Electronic and Information Engineering
The Hong Kong Polytechnic University, Hong Kong
e-mail: ron.li@connect.polyu.hk, enwcsiu@polyu.edu.hk, enpklun@polyu.edu.hk

***Abstract*—This paper presents a key frame recognition algorithm, using novel offline feature-shifts approach and search window weights. We extract effective feature patches from key frames with an offline feature-shifts approach for real-time key frame recognition. We focus on practical situations in which blurring and shifts in viewpoints occur in our dataset. We compare our method with some conventional keypoint-based matching methods and the newest CNN features for scene recognition. The experimental results illustrate that our method can reasonably preserve the performance in key frame recognition when comparing with methods using online feature-shifts approach. Our proposed method provides larger tolerance of unmatched pairs which is useful for decision making in real-time systems. Moreover, our method is robust to illumination and blurring. We achieve 90% accuracy in a nighttime sequence while CNN approach only attains 60% accuracy. Our method only requires 33.8 ms to match a frame on average using a regular desktop, which is 4 times faster than CNN approach with only CPU mode.**

Keywords—Key frame identification, vehicle detection, autonomous driving, visual place and key frame recognition

I. INTRODUCTION

Robot localization has been an overwhelming research topic in recent decades because of the rapid development of self-driving cars [1, 2]. Global Positioning System (GPS) is widely used in various commercial localization systems but satellite signals are generally affected by reflection and masking due to the concrete buildings, dense trees, etc [1]. Other different kinds of sensors such as wheel odometer and inertial sensor [2] are also solutions to the localization problem. Nevertheless, many of them are costly and also have their respective limitations. Under the situations, single camera based approach plays a crucial role in robot localization systems as its richness of information and cost-effectiveness. For single vision-based localization systems [3]-[12], different kinds of features are used for measuring the similarity between templates and query frames so as to report the best match to each query frame. Apart from the scene recognition module, there could be other building blocks such as vehicle detection [13, 14], and front car distance estimation [15] in localization systems. Therefore, the time cost of each building block algorithm is a critical issue as the entire system should be in real-time but not just for a specific module.

For all detection and recognition tasks [3]-[14], feature extraction is a standard procedure in localization algorithms. Artificially designed features and relative matching or evaluation methods are application-oriented for acquiring satisfactory results under some hypotheses. We have always to balance the time cost of the algorithm against the confidence in making final decisions. For example, [3] combined BRIEF and Gist descriptors into BRIEF-Gist descriptor for scene recognition using Hamming distance. [4] proposed an appearance-based approach to Visual Simultaneous Localization and Mapping (Visual SLAM) using only low-resolution images. [5] is an improved version of [4] which added a patch-based verification process for refining the place matches. However, they employed parallel programming in order to deal with the heavy computation requirement.

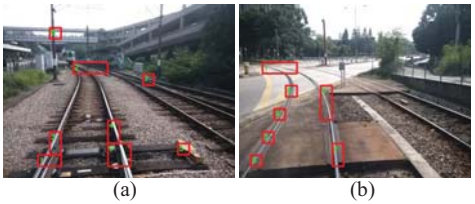
[7, 8] suggested that there are straight paths where the scenes cannot be discriminated effectively. [7] proposed a key frame approach to tackle the problem. Scenes with high discrimination power can act as key frames for high confidence matching and to lock the current location of the train and perform tracking for less possible scenes.

In recent years, many research teams have applied CNN features to their methods so as to enhance the adaptability to changes in conditions and appearance. [9] assessed the performance of CNN features for scene recognition with AlexNet [16] and discovered that the AlexNet conv3 layer features provide the best performance in localization tasks. Subsequently, [10] fused features from different layers together. [11] suggested to learn the representative features from different types of features. Their results reflected that Histograms of Oriented Gradients (HOG) [17] and CNN features occupied the main section of the learned representative features, from 83.1% to 95.1%. [12] modelled the localization problem as a network flow problem. They used HOG and AlexNet conv3 layer features to evaluate their method and found that two types of features could achieve similar performance in some situations.

In this paper, we consider that key frames [7] have already been extracted from a sequence of reference frames. Frames with high discrimination power in a reference sequence have been extracted as key frames for key scene recognition. We assume that the key feature patches (KFPs) which compose of one or more blocks of the key frames have also been extracted by comparing with all other frames in the reference sequence as shown Fig.1 and these patches are stored in HOG patch format with standard HOG feature vector formation [17].

The major contribution of this paper is that we propose an offline feature-shifts approach which greatly reduces the complexity of the recognition process comparing with online feature-shifts approach in conventional matching procedures [5, 18]. Also, with a weighting approach based on the HOG patch-based matching distance, the discrimination power of key

frames can largely be enhanced and be comparable to or even better than that of using the CNN approach.



(a) (b)

Fig.1. (a) Key frame, Frame 403, from LRT dataset with 8 key feature patches (b) Key frame, Frame 2533, from LRT dataset with 7 key feature patches

The rest of this paper is organized as follows. Section II describes our proposed method for offline feature-shifts approach and online key frame recognition with weighting approach. Section III gives experimental results and comparisons of various approaches. Finally, Section IV provides a conclusion of the paper.

II. PROPOSED METHOD

Key frames are extracted if these frames contain distinguished features different from other frames. Key feature patches (KFPs, we also refer KFP as patch later) with fixed or variable sizes (in terms of basic key feature blocks, KFBs) are identified within a key frame, such that these form the most distinguished patches within the key frame. Usually there are something like 5 to 12 KFPs for a key frame. The extraction of Key frames, KFBs and KFPs are outside the scope of this paper. Fig.2 is a simplified flow chart of key frame identification process.

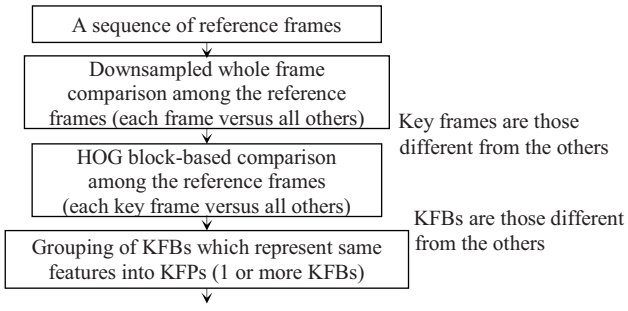


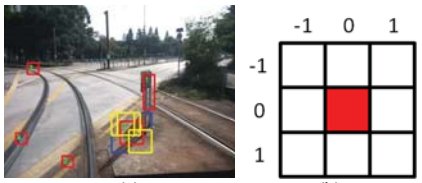
Fig.2. Flow chart of Key Frame Identification

A. Offline Feature-Shifts Approach

The objective of this step is to reduce the time cost of real-time key frame recognition. In conventional feature-based matching procedures [5, 18], we match a feature in a query frame by building a search window and search for the location which gives the highest similarity between features. This is what we call online feature-shifts approach as we shift the features in the query frame. The time cost of the online feature-shifts approach involves the comparison of two feature vectors and also the formation of shifted feature vectors which occupies most of the time cost. We propose to shift the key feature patches in an offline manner for predicting the feature-shifts in the query frame. The proposed offline feature-shifts approach is illustrated as follows.

Suppose that we set the search range to [-a, a] pixels and the stride is s pixels. The total number of shifted versions of a patch, (T_s) is calculated as:

$$T_s = (\frac{2a}{s}+1)^2 \quad (1)$$



(a) (b)

Fig.3. (a) The proposed offline feature-shifts approach (red-bounded regions: KFPs with their initial locations, blue-bounded region: search window for current KFP, yellow-bounded regions: shifted versions of the current KFP) (b) Illustration of a search window

Each shifted version is indicated by (m, n) where m and n are the horizontal and vertical shifts respectively, as shown in Fig.3b, with a = 1 and s = 1. We have to record the HOG feature vector for each shifted version of the patches and store it into a database for online usage. Note that the initial location of each patch in a key frame has also to be stored with the computed feature vectors.

B. Key Frame Recognition with Weighting Approach

For online key frame recognition, we calculate the feature vector of each patch in an incoming query frame based on the initial locations of the patches of a key frame. The feature vector of each patch in the query frame is compared with T_s versions of feature vectors stored in the database (reference key frames with key feature patches) using Cosine similarity.

$$C(\mathbf{p},\mathbf{q}) = \cos\omega = \frac{\mathbf{p}\cdot\mathbf{q}}{\|\mathbf{p}\|\|\mathbf{q}\|} \quad (2)$$

where cosω is the Cosine similarity which is defined as the cosine of the angle difference between vectors **p** and **q**.

For each patch, we find out the best matched location among all the shifted versions which gives the highest similarity.

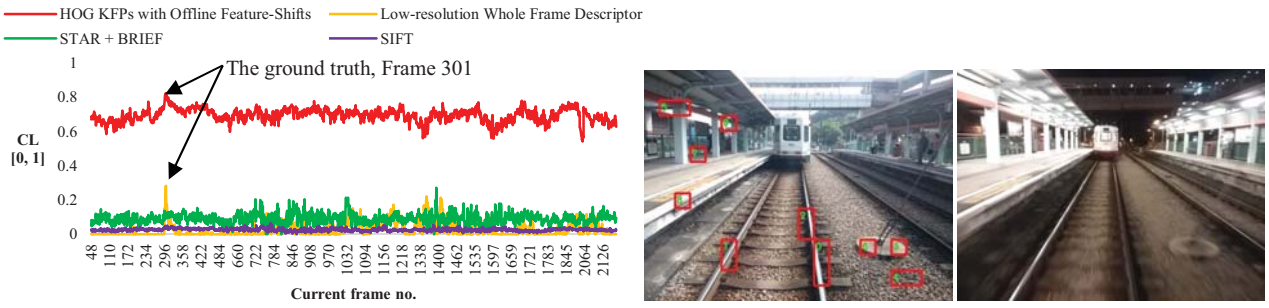
$$(\Delta m_{k,\min}, \Delta n_{k,\min}) = \underset{\Delta m_k, \Delta n_k \in [-\frac{a}{s}, \frac{a}{s}]}{\arg\max}\ C(F_{k,query}, F_{k,key,\Delta m_k,\Delta n_k}) \quad (3)$$

where $F_{k,query}$ is the k^{th} patch in the query frame and $F_{k,key,\Delta mk,\Delta nk}$ is the shifted version of the k^{th} KFP in the key frame at (Δm_k, Δn_k). We target at ($\Delta m_{k,\min}$, $\Delta n_{k,\min}$) which is the best match to the k^{th} KFP with Cosine similarity, C_k.

$$C_k = C(F_{k,query}, F_{k,key,\Delta m_{k,\min},\Delta n_{k,\min}}) \quad (4)$$

In order to ensure that the patches in the query frame are matched to the original KFPs we found previously, each patch is weighted by the distance from the initial location of its corresponding KFP in the comparing key frame.

Fig.5 shows a search window with m and n range from -3 to 3. $d_{\max}$ is the largest L2 distance [17] inside the search window, sqrt(18) = sqrt(3^2+3^2) in this case. Based on $d_{\max}$, we normalize

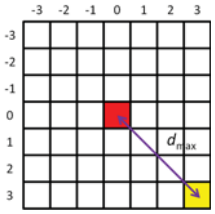


(a) (b) (c)

Fig.4. (a) Discrimination power of our HOG KFPs approach and other 3 conventional approaches in key frame recognition (b) A key frame, Frame 571, from LRT dataset with 10 KFPs (c) The ground truth of the key frame in the current sequence, Frame 301 (under changes in illumination and blurring)

the distance inside the search window and the weighting for the k^{th} patch, w_k, can be computed.

$$w_k = 1 - \frac{d_{(\Delta m_{k,\min}, \Delta n_{k,\min})}}{d_{\max}} \tag{5}$$



Red: Initial location of a key feature patch (KFP)

Yellow: The farthest location from the origin

Fig.5. A search window with m and n range from -3 to 3

where we compute the distance between ($\Delta m_{k,\min}$, $\Delta n_{k,\min}$) and the origin of the k^{th} KFP. If the distance is very small, the patch is matched to what we found originally and we trust the matching result by multiplying a larger weighting. Otherwise, C_k is hugely lowered. We calculate the overall confidence level (CL) between two frames.

$$CL = \frac{\sum_{k}^{K} C_k w_k}{K} \tag{6}$$

where K is the number of KFPs in a key frame, C_k and w_k are calculated using Eqn. (4) and (5) respectively.

III. EXPERIMENTAL RESULTS

A. Dataset

A large amount of experiments have been done. We have compared our method with 4 other approaches, namely STAR detector [19] with BRIEF descriptor [20], SIFT [21], low-resolution whole frame descriptor [4], and AlexNet conv3 layer features based approach [9, 12]. The first two are the keypoint-based approaches in which detector is used to detect keypoints in a frame and a descriptor is used to describe each of the keypoints. BRIEF is a binary descriptor while SIFT describes each keypoint in a real value format. Low-resolution whole frame descriptor is a global descriptor which compares the structural features between frames. For CNN approach, a pre-trained AlexNet [22] replaces the conventional feature extraction process and we used the AlexNet conv3 layer features as features for similarity measure. Our dataset is acquired from a public transportation in Hong Kong, Light Rail Transit (LRT) which consists of 4 sequences of the same route, 3 in the daytime and 1 at nighttime. We used one daytime sequence as a reference sequence for key frame identification. This dataset involves practical issues such as extreme lighting conditions and blurring. Each of the four sequences contains 2565 images on average.

B. Feature Implementation and Hardware Details

All approaches used in evaluation were implemented by C++ programming language, except the AlexNet conv3 layer features based approach. STAR detector [19] with BRIEF descriptor [20], SIFT [21], and low-resolution whole frame descriptor [4, 5] were implemented using OpenCV lib. 2.4.13. The AlexNet conv3 layer features [9, 12] were extracted under the Caffe framework [23] pre-trained by ImageNet [22]. For the hardware, i7-4790 CPU and GTX 1080 GPU were used for time cost evaluation. Note that no code optimization technique and parallel programming has been-applied.

C. Conventional Approaches on Key Frame Recognition

For both keypoint-based scene recognition methods, we used the conventional keypoints matching procedures [21] for evaluating the similarity between two frames. We matched the keypoints in a key frame to a query frame using Hamming distance for binary descriptor (BRIEF) or L2 distance for real value descriptor (SIFT). We also define that there is a "good match point" only if the distance from a keypoint to a proposed match point is at least 20% smaller than the distance of all other proposed match points. The confidence level (CL) of a frame to a key frame is the ratio of the good match points to the total number of keypoints of that key frame. For low-resolution whole frame approach, we used average pixel difference to calculate the distance between two frames which is bounded between 0 to 255. In our experiments, we found that all the distances between frames always range from 50 to 100. Therefore, we bounded the distance between two frames from 50 to 100 manually and convert it into CL which ranges from 0 to 1 for concise comparison of different approaches. Fig.4a shows the discrimination power of our HOG key feature patches (KFPs) approach and other 3 conventional approaches. Its corresponding key frame is shown as Fig.4b and we match this key frame with a nighttime sequence. As indicated by the arrows in Fig.4a, it is clear that only low-resolution whole frame approach and HOG KFPs with offline feature-shifts approach give the correct match pair (peak of the curve, frame 301 of the current sequence which is shown in Fig.4c). Similar observation is found for the cases of other sequences. The results show that our HOG KFPs with offline feature-shifts approach performs better compared with other approaches with

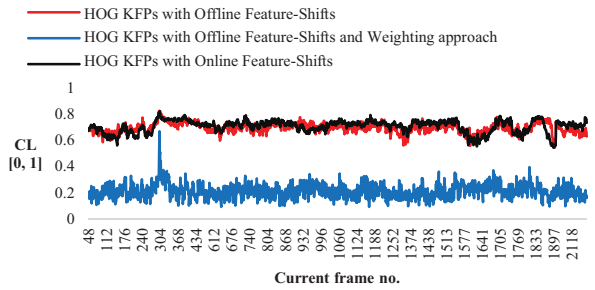


Fig.6. Discrimination power of our HOG key feature patches with offline or online feature-shifts and weighting approach

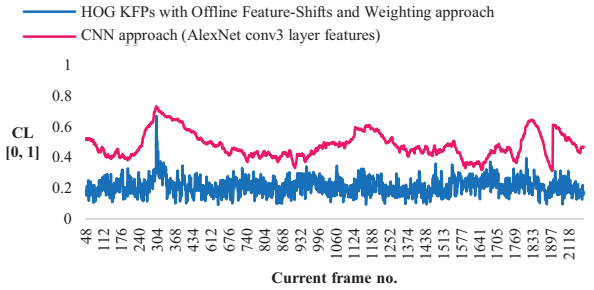


Fig.7. Discrimination power of our proposed HOG KFPs with offline feature-shifts and weighting approach, and CNN approach

TABLE I. Overall Accuracy and Tolerance of our proposed method and other approaches

Testing sequence	Accuracy (%)						Tolerance					
	Low-resolution whole frame	STAR+ BRIEF	SIFT	Proposed method	HOG patches with online shifts	CNN approach	Low-resolution whole frame	STAR+ BRIEF	SIFT	Proposed method	HOG patches with online shifts	CNN approach
LRT2-Night	0.7	0.0	0.1	**0.9**	0.8	0.6	0.256	0.0	0.033	**0.389**	0.259	0.22
LRT3-Day	0.8	0.8	**0.9**	**0.9**	**0.9**	**0.9**	0.301	0.257	0.121	0.311	**0.355**	0.302
LRT4-Day	0.9	0.8	**1.0**	0.9	**1.0**	**1.0**	0.339	0.163	0.056	0.324	**0.34**	0.297

extreme lighting and blurring conditions. However, there should be rooms of improvement as shown below.

D. HOG Key Feature Patches with Weighting Apporach

Fig.6 shows the discrimination power of HOG KFPs approach with our offline or conventional online arrangements and our weighting approach. The key frame and matching scenario are the same as discussed previously. With our proposed weighting approach, the tolerance of the key frame recognition is greatly enhanced. We define the tolerance as the difference between the peak *CL* and the average *CL* of a curve. Large tolerance means that we have a larger decision boundary which is important to temporal informative localization problems. Considering the tolerances of the red and blue curves, there is an increase of 0.33, from 0.13 to 0.46. Also, the average matching time of this key frame has only increased from 37.17 ms to 41.35 ms. Note also that this key frame has 10 KFPs (see Fig.4b) which consist of 16 KFBs. This means that 10 weightings are included and our weighting approach only costs an extra of a few milliseconds.

E. Online and Offline Feature-Shifts Approach

From Fig.6, it is obvious that the performance of offline and online feature-shifts approaches without our weighting approach are similar. The tolerances of the red and black curves are 0.13 and 0.11 respectively. However, the average matching time is 405 ms if online feature-shifts approach is used. It is nearly 10 times on average slower when compared with the matching time using our proposed method. Note that we considered 169 (with $a = 24$ and $s = 4$ in Eqn. (1)) shifted-patches from an anchor patch.

F. Comparing CNN Approach for Key Frame Recognition

Fig.7 also shows the discrimination power of CNN approach and our proposed method. We observe that the tolerances of the blue and pink curves are 0.46 (our) and 0.25 (CNN) respectively. The dimension of AlexNet conv3 layer features based approach is 384x13x13 = 64896 [9, 16]. If GPU is used, the average feature extraction time is 3.41 ms. If only CPU is used, 133.44 ms is required to extract features on average. The feature matching time of a 64896-length feature vector is less than 0.5 ms. Our proposed method is 3 times faster than CNN approach using CPU mode for this typical example of our study.

G. Overall Comparison of Various Features

Table I shows the accuracy and tolerance comparisons of different approaches. For the LRT sample dataset in this paper, there are 10 key frames. The average numbers of key feature patches (KFPs) and key feature blocks (KFBs) per key frame are 8.5 and 13.3 respectively. If the difference between the reported best match and the ground truth is less than 10 frames, we regard the match as a correct match. For our offline feature-shifts approach, we set $a = 24$ and $s = 4$, hence 169 versions.

We can observe that our proposed method has good performance in the nighttime sequence, and all approaches perform similarly in the daytime sequences. [24] claimed that blurring is a limitation of the CNN trained on ImageNet. Our results also reflect this problem as the nighttime sequence suffers from the problem of blurring, as shown in Fig.4c. For tolerance comparison, we calculate the tolerance only if the reported best match is a correct match. We can see that our proposed method can provide a higher tolerance in general.

TABLE II. Overall time cost of our proposed method and other approaches

Testing sequence	Time cost (ms)					
	Low-resolution whole frame	STAR + BRIEF	SIFT	Proposed method	HOG patches with online shifts	CNN approach
LRT2-Night	**0.0119**	13.4	386.2	34.2	322.02	CPU: 134 GPU: 3.6
LRT3-Day		14.1	432.0	33.59	340.73	
LRT4-Day		14.6	452.9	33.6	311.01	

Without the use of GPU, the time cost of CNN approach and HOG patches with online feature-shifts approach are too high and not applicable to real-time systems. From Table II, CNN approach with CPU mode requires 134 ms per frame on average. The fastest method is the low-resolution whole frame approach which only requires 0.0119 ms per frame on average. For our proposed method, 33.8 ms is required for one frame matching on average and the required computation time is still short which is acceptable to real-time systems.

IV. CONCLUSION

In this paper, we have proposed a novel method which employs offline feature-shifts and weighted patches approach. With our proposed method, the performance in key frame recognition is very fast when comparing with the conventional online feature-shifts approaches. Also, our proposed method can provide larger tolerance in general which is useful for decision making. In terms of time cost, our proposed method is applicable to real-time systems because of the use of offline feature-shifts approach. On average, our proposed method is 4 times faster than CNN approach in CPU mode. Therefore, our proposed method is suitable for key frame recognition in localization problems. For future development, we will try to combine the low-resolution whole frame descriptor and our method together for enhancing further the performance in a localization system. Low-resolution whole frame descriptor is a global descriptor which describes the structural features of a frame. Our method focuses on key feature patterns which are effective local descriptors to describe a key frame. We believe that we can fuse these two features together and provide superior localization results in our future studies.

ACKNOWLEDGMENT

This work was supported in part by the Hong Kong Polytechnic University (1-ZE1B), and Mr. Li Chu Tak would like to acknowledge the Postgraduate Studentship offered by the same university.

REFERENCES

[1] Guillaume Bresson, Zayed Alsayed, Li Yu, and Sébastien Glaser, "Simultaneous Localization and Mapping: A Survey of Current Trends in Autonomous Driving," *IEEE Transactions on Intelligent Vehicles*, vol. 2, no. 3, pp. 194-220, Sept. 2017.

[2] Cesar Cadena, Luca Carlone, Henry Carrillo, Yasir Latif, Davide Scaramuzza, José Neira, Ian Reid, and John J. Leonard, "Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age," *IEEE Transactions on Robotics*, vol. 32, no. 6, pp. 1309-1332, Dec. 2016.

[3] Niko Sünderhauf and Peter Protzel, "BRIEF-Gist - Closing the Loop by Simple Means," *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, San Francisco, CA, USA, pp. 1234-1241, Sep. 2011.

[4] Michael J. Milford and Gordon F. Wyeth, "SeqSLAM: Visual Route-Based Navigation for Sunny Summer Days and Stormy Winter Nights," *Proceedings, IEEE International Conference on Robotics and Automation (ICRA)*, Saint Paul, MN, USA, pp. 1643-1649, May 2012.

[5] Michael J. Milford, Walter Scheirer, Eleonora Vig, Arren Glover, Oliver Baumann, Jason Mattingley, and David Cox, "Condition-Invariant, Top-Down Visual Place Recognition," *Proceedings, IEEE International Conference on Robotics and Automation (ICRA)*, Hong Kong, China, pp. 5571-5577, May 2014.

[6] Yongliang Qiao, Cindy Cappelle, and Yassine Ruichek, "Visual Localization across Seasons using Sequence Matching based on Multi-Feature combination," *Journal of Sensors*, vol. 17, no. 11, pp. 2442-2463, Oct. 2017.

[7] Meng Yao, Wan-Chi Siu and Ke-Bin Jia, "Learning-based Scene Recognition with Monocular Camera for Light-Rail System," *Proceedings*, *IEEE International Conference on Industrial Electronics for Sustainable Energy Systems (IESES)*, Hamilton, New Zealand, pp.230-236, 30 Jan. to 2 Feb. 2018.

[8] Rafael Peixoto Derenzi Vivacqua, Massimo Bertozzi, Pietro Cerri, Felipe Nascimento Martins, and Raquel Frizera Vassallo, "Self-Localization based on Visual Lane Marking Maps: an Accurate Low-Cost Approach for Autonomous Driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 2, pp. 582-597, Feb. 2018.

[9] Niko Sünderhauf, Ssareh Shirzai, Feras Dayoub, Ben Upcroft, and Michael Milford, "On the Performance of ConvNet Features for Place Recognition," *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Hamburg, Germany, pp. 4297-4304, Sept. 2015.

[10] Roberto Arroyo, Pablo F. Alcantarilla, Luis M. Bergasa, and Eduardo Romera, "Fusion and Binarization of CNN Features for Robust Topological Localization across Seasons," *Proceedomgs, IEEE/RSJ Intermational Conference on Intelligent Robots and Systems (IROS)*, Daejeon, South Korea, pp. 4656-4663, Oct. 2016.

[11] Fei Han, Xue Yang, Yiming Deng, Mark Rentschler, Dejun Yang, and Hao Zhang, "SRAL: Shared Representative Appearance Learning for Long-Term Visual Place Recognition," *IEEE Robotics and Automation Letters (RAL)*, vol. 2, no. 2, pp. 1172-1179, Apr. 2017.

[12] Tayyab Naseer, Wolfram Burgard and Cyrill Stachniss, "Robust Visual Localization Across Seasons," *IEEE Transactions on Robotics*, vol. 34, no. 2, pp. 289-302, Apr. 2018.

[13] Xue-Fei Yang and Wan-Chi Siu, "Vehicle Detection under Tough Conditions using Prioritized Feature Extraction with Shadow Recognition," *Proceedings, IEEE 22nd International Conference on Digital Signal Processing (DSP)*, London, UK, pp. 1-5, Aug. 2017.

[14] Chup-Chung Wong, Wan-Chi Siu, Paul Jennings, Stuart Barnes, and Bernard Fong, "A Smart Moving Vehicle Detection System Using Motion Vectors and Generic Line Features," *IEEE Transactions on Consumer Electronics*, vol. 61, no. 3, pp. 384-392, Aug. 2015.

[15] Hoi-Kok Cheung, Wan-Chi Siu, Steven Lee, Lawrence Poon, and Chiu-Shing Ng, "Accurate Distance Estimation Using Camera Orientation Compensation Technique for Vehicle Driver Assistance System," *Proceedings, IEEE International Conference on Consumer Electronics (ICCE)*, Las Vegas, NV, USA, pp. 231-232, Jan. 2012.

[16] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *Proceedings, the 25th International Conference on Neural Information Processing Systems (NIPS),* Lake Tahoe, Nevada, USA, pp. 1097-1105, Dec. 2012.

[17] Navneet Dalal and Bill Triggs, "Histograms of Oriented Gradients for Human Detection," *Proceedings, IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, San Diego, CA, USA, pp. 886-893, Jun. 2005.

[18] Stephanie Lowry, Niko Sünderhauf, Paul Newman, John J. Leonard, David Cox, Peter Corke, and Michael J. Milford, "Visual Place Recognition: A Survey," *IEEE Transactions on Robotics*, vol. 32, no. 1, pp. 1-19, Feb. 2016.

[19] Motilal Agrawal, Kurt Konolige, and Morten Rufus Blas, "CenSurE: Center Surround Extremas for Realtime Feature Detection and Matching," *Proceedings, the 10th European Conference on Computer Vision (ECCV)*, Marseille, France, pp. 102-115, Oct. 2008.

[20] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua, "BRIEF: Binary Robust Independent Elementary Features," *Proceedings, the 11th European Conference on Computer Vision (ECCV)*, Hersonissos, Heraklion, Crete, Greece, pp. 778-792, Sept. 2010.

[21] David G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91-110, Nov. 2004.

[22] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei, "Imagenet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211-252, Dec. 2015.

[23] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell, "Caffe: Convolutional Architecture for Fast Feature Embedding," *Proceedings, the 22nd ACM International Conference on Multimedia*, Orlando, Florida, USA, pp. 675-678, Nov. 2014.

[24] Philipp Fischer, Alexey Dosovitskiy, and Thomas Brox, "Descriptor Matching with Convolutional Neural Networks: a Comparison to SIFT," *arXiv preprint arXiv:1405.5769v1*, pp. 1-9, May 2014.