Classification of SURF Image Features by Selected Machine Learning Algorithms

Karel Horak, Jan Klecka, Ondrej Bostik and Daniel Davidek
Brno University of Technology
Brno, Czech Republic
Email: {horak,klecka,bostik,davidek}@feec.vutbr.cz

***Abstract*—We have proposed a concept for classification interesting points in images by means of a machine learning approach. The basic idea is that each interesting point detected in an image is classified either as a point belonging to some trained model (e.g. corner of a license plate) or not. During the first stage, we detected interesting points in a set of images by the well-known SURF method. Then we have employed supervised learning algorithms LDA, QDA, Naive Bayes, Decision tree and SVM to create relevant models of corners in images. Finally, all generated models were evaluated during classification stage by a cross-validation technique and an example experiment of license plate detection has been carried out and is introduced at the very end of this paper. Interesting outcomes have been obtained by the Naive Bayes algorithm resulting in a sensitivity value of the 100% and an accuracy value of the 99.8% on the real-world gallery of 535 images containing over 93 thousand interesting points. Although our gallery is not vast, the results are really promising to use our concept in another applications of robust and real-time object recognition.**

***Keywords*—corners; image recognition; interesting points; license plate; supervised learning; SURF;**

I. Introduction

Machine learning can be comprehended as a part of an artificial intelligence discipline concerned with a design and implementation of algorithms that can learn autonomously. Expert systems, neural networks [1], genetic algorithms [2], differential evolution, data mining programs and others [3] are all examples of theories often implemented by means of the machine learning. Approaches in the machine learning domain can be divided into the three groups: supervised learning, unsupervised learning and so-called semi-supervised learning [4]. A difference between these groups are given by a structure of an input dataset. If the dataset contains an information about actual class of a measured data (sometimes referred as observations) in addition to these data, we speak about the supervised learning. On the other hand, it is a task of the unsupervised learning if we have not the information about actual class (e.g. classification label) of the input data. The semi-supervised learning means the input dataset (training signal) is incomplete so that only some of input examples are equipped with their classification label or regression value.

The completion of this paper has been supported by the grant No. FEKT-S-17-4234 - Industry 4.0 in Automation and Cybernetics under the Internal Science Fund of Brno University of Technology.

Because the most of the real-world tasks are either hard to model explicitly, NP-hard or only poorly defined, the machine learning methods try to use statistical reasoning to find approximate solutions for tackling these difficulties. A classification of generic features detected in images is a representative example of the first option. Typical attribute of a recognition task in computer vision is that the input dataset is provided with an autonomous or manual labelling machine but the explicit definition of the desired objects detected is either highly challenging or even impossible. As an example we can introduce our own task of a vast image gallery of vehicles with license plates (hereinafter referred as the LPs) and a file with annotations i.e. list of license plates corners. We are able to design and implement a machine learning algorithm to solve LPs localization task but we are not able explicitly define the term ”corner of the license plate” in sense of image pixels.

II. State-of-the-art: Machine Learning in Computer Vision

A. Supervised Learning

The aim of supervised machine learning is to build a model that makes predictions based on evidence in the presence of uncertainty [5]. As adaptive algorithms identify patterns in data, a computer learns from the observations. When exposed to more observations, the computer improves its predictive performance [6]. Specifically, a supervised learning algorithm takes a known set of input data and known responses to the data (output), and trains a model to generate reasonable predictions for the response to new data. A basic workflow of the supervised learning is shown in Fig. 1. During a learning

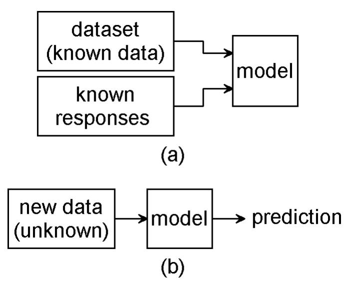

Fig. 1. Workflows of supervised learning (a) and prediction stage (b).

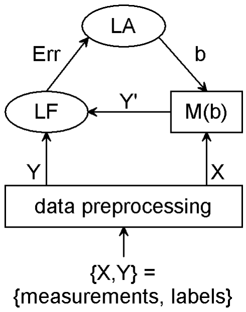

Fig. 2. General scheme of the learning stage process.

stage (a) the input dataset and known responses are used to train a model. The other stage of prediction (b) evaluates a new data of unknown response to predict it by the model.

As is depicted in the Fig. 2, consider symbols `X` and `Y` be a measured data (observations) and known responses respectively. The learning stage is then iterative process of finding parameters `b` of the model `M` based on an error value `Err`. Meaning of all symbols used in the figure above is following:

`X` = measurements (observations)
`Y` = actual class (correct label)
`Y'` = predicted class (estimated label)
`Err` = error based on the difference between Y and Y'
`LF` = loss function
`LA` = learning algorithm
`M(b)` = model with parameters b

The iterative learning process is terminated when the `Err` value descends under a predefined limit and then the trained model can be used for a prediction of the unknown data.

To understand what the term prediction means a definition of a classification and regression follows [7]. Supervised learning splits into two broad categories: classification and regression. A difference between these two terms is in a type an output value. The regression results in continuous value e.g. scalar value whereas the classification means labelling an unknown input data into finite discrete classes. In other words, in classification the goal is to assign a class (or label) from a finite set of classes to an observation whilst in regression, the goal is to predict a continuous measurement for an observation. We only consider the classification option in this paper [8].

B. Overview of Interest Points Detectors

Because we have employed a concept of an interest point in an image as a part of our classification method, we provide a short overview of the most often used interest points detectors here. In the literature, a plenty of various interest points detectors and region descriptors can be found out. We enumerate several of the most influential here in chronological order. At the very beginning, Harris and Stephens introduced their so-called Harris operator for corner detection in [9] in 1988. Several years later, Shi and Tomasi introduced their concept of feature quality based on eigenvalues of a local window in an image Eigen in [10] in 1994. Significant quality step in feature extraction was made by Matas et al. by introducing their idea of a Maximally Stable Extremal Region (MSER) in [11] in 2002. Well-known and very often used Scale Invariant Feature Transform (SIFT) was introduced by David Lowe in [12] in 2004. Next, Mikolajczyk and Schmid suggested an extension of the SIFT descriptor named Gradient Location and Orientation Histogram (GLOH) in [13] in 2005. Further, Rosten and Drummond introduced a relatively novel idea of very fast and efficient feature point detection called Features from Accelerated Segment Test (FAST) in [14] in 2005. Next chronologically, Dalal and Triggs introduced their Histogram of Oriented Gradients (HOG) in [15] in 2005 and Bay et al. introduced their interesting theory of Speeded-Up Robust Features (SURF) in [16] in 2008. After this computational milestone, several new detectors and derived alternatives to previously defined feature point ideas were introduced by Calonder et al. (BRIEF in [17] in 2010), by Leutenegger et al. (BRISK in [18] in 2011, by Rublee et al. (ORB in [19] in 2011) and by Alahi et al. (FREAK in [20] in 2012). Finally, the interesting approach of a deep learning named as LIFT (Learned Invariant Feature Transform) combining a detector, orientation estimator and descriptor into one pipeline process was introduced recently by Yi et al. in [21].

From the list above we have picked out the SURF method for the interesting points detection in an example below. Nevertheless our approach is independent of this choice and any of the above listed detectors (and others unmentioned) can be used as a generator of the input dataset.

III. Classification of Image Features - Concept

We have designed a concept of multidimensional classification of features in images based on several selected supervised machine learning algorithms. The features in images are for example interesting points or regions detected in an image by one of many known detectors as Moravec [22], Harris [9], Shi&Tomasi [10], FAST [14], SIFT [12], MSER [11], SURF [12], etc. As already mentioned we have employed the SURF method convenient for real-time implementations.

Consider a concept workflow introduced in the Fig. 3. We have a set of images $f_i(x, y), i = 1, ..., N$ which are used as an input for an interesting points detector. The SURF algorithm detects a set of pixels in each input image. All these pixels fulfil conditions to be a corner, i.e. a high image gradient in the both vertical and horizontal directions. By means of brightness values in a neighbourhood of each interesting point

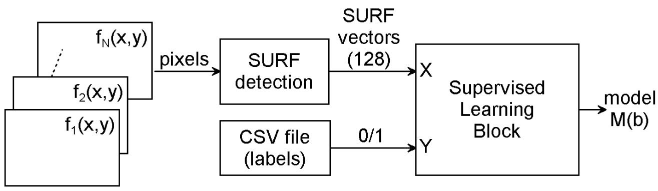

Fig. 3. Proposed workflow of the learning stage.

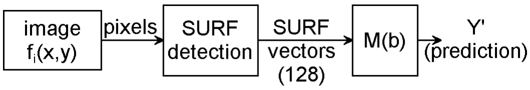

Fig. 4. Proposed workflow of the classification stage.

a vector X of 128 scalar values is computed. This vector unambiguously describes properties of the interesting points in the sense of an affine transformation (orientation, scale, skew). Besides, annotation labels Y (e.g. in form of CSV file) representing known responses has been manually or semi-manually created. Note that this manual annotation is a part of almost all supervised learning tasks and constitutes the main drawback of this approach because of time demanding process. A so-called *deep learning* concept arises in the last few years to avoid this annoying and often imprecise step.

As for annotation labels, a value of 1 is related to corners representing an object to detect (positive examples, e.g. corner of a license plate) and vice versa a value of 0 describes the others (negative examples, e.g. corner detected on a vehicle or in surround). Based on the input training data {X,Y} the model M(b) is generated by the supervised learning block. This block can be implemented by any of supervised machine learning algorithms. As the supervised learning block, we have examined the *Linear Discriminant Analysis* (LDA), *Quadratic Discriminant Analysis* (QDA), *Naive Bayes* (NB), *Decision Tree* (DecTree) and the *Support Vector Machines* (SVM) algorithms during our experiment described in the next chapter.

In the Fig. 4 the classification stage is depicted. A new "unknown" image $f_i(x, y)$ is submitted to the SURF interesting points detector and a prediction (estimation) Y' of detected corners are evaluated by the model M(b). All corners with prediction value of 1 are subsequently processed as a part of desired object and the others are ignored.

IV. Classification of Image Features - Example

We have tested the above introduced concept of classification interest points on our own image gallery of various vehicles. The aim of this task is to localize a license plate of a car by a detection its four corners (upper left and right and lower left and right). Earlier, we proposed another method called IPDES (*Interest Point Detector of Expected Structure*) in [23]. This approach was based on seeking a certain geometrical structure of the detected interest points. It means an explicit model of the desired object was known or expected (horizontal rectangle). Moreover, the IPDES method was computationally demanding due to many detected corners in an image and an achieved precision was not sufficiently high (the true positive ratio slightly above 95%). Our current approach based on the machine learning procedures does not require the explicit model of the LPs corners and moreover reaches higher precision and accuracy as will be shown in the next chapters.

A. Experimental Workflow

We have employed our suggested workflow for the learning and classification stage depicted in the Fig. 3 and the Fig. 4 respectively. Predicted value is either of class LP for interesting points belonging to one of four license plate corners or of class Not LP for every other corners.

Moreover, we implemented the *Naive Bayes* algorithm in two modifications: at first, the predictor distribution within each class using a Gaussian distribution having given mean and standard deviation (model referred as the NB). The other implementation uses categorical predictor, when variables distribution is multivariate, multinomial random (model referred as the NB MV). At the classification (prediction) stage the cross-validation method has been employed to process the input dataset. To compare all trained models a so-called resubstitution error, learning time and a total of five other parameters were obtained during experiments and computed from confusion matrices.

B. Image Dataset

A dataset is represented by our own gallery of the 535 images. Only one car is present in each image, nevertheless it is not a required condition for any algorithm or processing step. Several selected examples of the images in the gallery can be seen in the Fig. 5. The whole gallery was used as the training dataset and the cross-validation method was used to evaluate an overall performance of each examined method (LDA, QDA, etc.). The cross validation splits the training data into 10 parts at random. Next, it trains 10 new models, each one on nine parts of the data and then examines the predictive accuracy of each new model on the data not included in training stage. This method gives a good estimate of the predictive accuracy of the resulting model, since it tests the new models on new data. Note that total of 93045 interesting points (corners) were detected in all images in the gallery. It means almost 200 corners were detected on each image on average and only 4 corners naturally relates to a single license plate (approx. 2% of all corners are "interesting").

Fig. 5. Examples of the images in the gallery as the input dataset.

		Predicted Class P	Predicted Class N
Actual Class	P	True Positives (TP)	False Negatives (FN)
	N	False Positives (FP)	True Negatives (TN)

Fig. 6. Structure of the confusion matrix with TP, FN, FP and TN values.

C. Models Evaluation Method

A confusion matrix is a specific table layout that allows visualization of the performance of an algorithm, typically a supervised learning one [24]. Each column of the matrix (see the Fig. 6) represents the instances in predicted classes while each row represents the instances in actual (known) classes. A plenty of various parameters of the trained model can be then computed on the basis of the true positive TP, true negative TN, false negative FN and false positive FP values.

We have selected the five most often used parameters for algorithms evaluation: sensitivity $Sens$ (true positive rate), specificity $Spec$ (true negative rate), precision $Prec$ (positive predicted value), accuracy Acc and so-called F1-measure $F1$ defined successively by the following equations (1) to (5). The last mentioned F1-measure is often explained as a harmonic mean of the $Sens$ and $Prec$ values. Comprehensive explanation of confusion matrix operations and a ROC analysis can be found in [25].

$$Sens = \frac{TP}{TP + FN}. \tag{1}$$

$$Spec = \frac{TN}{TN + FP}. \tag{2}$$

$$Prec = \frac{TP}{TP + FP}. \tag{3}$$

$$Acc = \frac{TP + TN}{TP + TN + FP + FN}. \tag{4}$$

$$F1 = \frac{2 \cdot TP}{2 \cdot TP + FP + FN}. \tag{5}$$

D. Experimental Results

The SURF detector has been applied on each image of the gallery to obtain a set of interesting points and their descriptors (128 values each). Detected interest points imprinted in a sample image are illustrated in the Fig. 7 by green circles. A feature vector `X` is computed for each detected SURF point and the predicted class `Y'` either `LP` or `Not LP` is determined by all the six models mentioned above. Confusion matrices of all evaluated models are represented by the Tables I up to VI.

The meaning of the values in the tables are following: the upper left value (TP) represents the interesting points correctly recognized as the corners of a license plate, and vice versa the lower right value (TN) represents the others interesting points correctly classified as the `Not LP` class.

Fig. 7. SURF interesting points (green circles) detected in a sample image and potential license plate regions (violet rectangles).

The two residual values on the anti-diagonal (FN and FP) represent misclassified corners. It means either corners actually not related to a license plate and wrongly classified to the `LP` class or corners actually related to a license plate and wrongly classified to the `Not LP` class. Generally, the true positive and true negative values (correct classifications on the main diagonal) should be maximized and on the other hand, the false positive and false negative values (wrong classifications on the anti-diagonal) should be minimized simultaneously.

By a quick sight at the tables, we can infer that the Naive Bayes Multivariate and the Decision Tree algorithms give best results because of the FP and FN values are really low. For precise comparison of all trained models we have aggregate results to the Table VII (confusion matrices parameters) and Table VIII (overall evaluation). For each investigated method (LDA, QDA, etc.) a value of the resubstitution error, learning time, sensitivity, specificity, precision, accuracy and F1-measure have been quantified.

As for the Table VII of confusion matrices parameters, higher number means better result. So we can see the Naive Bayes Multivariate and the Decision Tree algorithms have the highest values in average in comparison with other methods. The *Sensitivity* value of 100% of the Naive Bayes Multivariate algorithm means that all actual corners of the license plates have been classified correctly as the `LP` class, or alternatively no actual corner was misclassified as the `Not LP` class. The *Accuracy* value of 99.80% of the same algorithm tells that only 0.2% of the `Not LP` corners have been classified incorrectly to the `LP` class. This error represents a so-called over-classification and is not critical in our case because all

TABLE I. CONFUSION MATRIX LDA

	LP	**Not LP**
LP	1661	411
Not LP	1475	89498

TABLE II. CONFUSION MATRIX QDA

	LP	**Not LP**
LP	1915	157
Not LP	2636	88337

TABLE III. CONFUSION MATRIX NAIVE BAYES

	LP	**Not LP**
LP	1715	357
Not LP	14333	76640

TABLE IV. CONFUSION MATRIX N. BAYES MV

	LP	**Not LP**
LP	2072	0
Not LP	186	90787

TABLE V. CONFUSION MATRIX DEC. TREE

	LP	Not LP
LP	1957	115
Not LP	50	90923

TABLE VI. CONFUSION MATRIX SVM

	LP	Not LP
LP	385	1687
Not LP	32	90941

detected regions of the license plates are usually filtered by a subsequent OCR algorithm not recognizing a meaningful structure of numbers in the region. Two examples of the false LPs regions caused by the over-classification phenomenon are depicted in the Fig. 7 as violet rectangles above the only one actual LP region.

As for the Table VIII, the *Resubstitution Error*, *Learning Time* and the *F1-measure* have been acquired and computed to assess the algorithms. The *Resubstitution Error* is defined as the misclassification error i.e. the proportion of misclassified observations on the training dataset. For example, approximately the 2% of the interesting points of the training dataset were classified incorrectly by the LDA algorithm (see first cell in the Table VIII). The *Learning Time* provides an additional comparison of computational demands during the learning stage. Finally, the *F1-measure* parameter gives a crucial sight on to overall performance of the algorithms. It is obvious that a overall performance of the Naive Bayes Multivariate and the Decision Tree algorithms outperforms all others and are comparable to each other. Nevertheless, the Naive Bayes Multivariate algorithm offers approximately six time faster learning stage in a computational demands point of view than the Decision Tree algorithm.

E. Comparison with Other Works

There exists a lot of research papers related to the topic of automatic license plate recognition (ANPR) in the field of computer vision. Only a few approaches, however, use interesting point detection for a license plate localization in combination with some mechanism how to process plenty of obtained interesting point in an image. Relatively simple and straightforward system for a license plate localization was introduced in [26]. Here a concept using the SURF interesting point detection and subsequent filtering of outliers by the RANSAC has been employed. License plates detection ratio is presented to be 95% in the $Sens$ (sensitivity) parameter meaning on the vast gallery of publicly accessible images/databases.

A slightly different technique also using SURF method is proposed in [27]. Here, the SURF and well-known Bag-of Words feature descriptors are combined together and subsequently clustered by the K-means clustering to localize the license plates region in an image. The described framework has been tested on Malaysian license plate dataset with resultant accuracy subtly under 91%. The last paper good to mention [28] presents a new approach for vehicle license plate detection using a fusion of unigram and bigram model using SURF descriptors. Both unigram and bigram models of corners are learning using Support Vector Machine technique. Presented results slightly overcome $Prec$ (precision) parameter of 98% on testing database of only 97 real-world images.

Although our suggested approach overcomes these results selected over the world (SURF+DecTree or SURF+Naive Bayes MV correspond to 99.8% of the sensitivity $Sens$), it is not still convenient and precise value enough to real-world implementation without any constraints or practical restrictions.

TABLE VII. EVALUATION PARAMETERS FROM CONFUSION MATRICES

	Sensitivity	Specificity	Precision	Accuracy
LDA	80.2%	98.38%	53.0%	97.97%
QDA	92.4%	97.10%	42.1%	97.00%
NB	82.8%	84.24%	10.7%	84.21%
NB MV	100.0%	99.80%	91.8%	99.80%
DecTree	94.4%	99.95%	97.5%	99.82%
SVM	18.6%	99.96%	92.3%	98.15%

TABLE VIII. COMPARISON OF ALL CLASSIFIERS

	Resub. Error	Learn. Time	F1-measure
LDA	0.0203	5.3%	63.8%
QDA	0.0300	5.3%	57.8%
NB	0.1579	3.2%	18.9%
NB MV	0.0020	4.0%	95.7%
DecTree	0.0018	25.0%	95.9%
SVM	0.0185	100.0%	30.9%

V. CONCLUSION

We have described a theory about several supervised learning machine algorithms and have performed the comparative example experiment in this paper. The concept of classification of the SURF interesting points by the machine learning approach has been established, implemented and verified. For a chosen example of the license plate detection in an image, the Naive Bayes Multivariate algorithm gave the best and promising results, tightly followed by the Decision Tree algorithm. Because of relatively high values of the parameters *Sensitivity* and *Accuracy* and thus also the *F1-measure*, the future work will be focused on a verification of our approach to another galleries (datasets) containing other types of generic objects to reveal potential hidden lacks.

REFERENCES

[1] J. Schmidhuber, “Deep learning in neural networks: An overview,” *Neural Networks*, vol. 61, pp. 85–117, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0893608014002135

[2] K. De Jong, “Learning with genetic algorithms: An overview,” *Machine Learning*, vol. 3, no. 2-3, pp. 121–138, oct 1988. [Online]. Available: http://link.springer.com/10.1007/BF00113894

[3] D. W. Aha, D. Kibler, and M. K. Albert, “Instance-based learning algorithms,” *Machine Learning*, vol. 6, no. 1, pp. 37–66, jan 1991. [Online]. Available: http://link.springer.com/10.1007/BF00153759

[4] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, ser. Springer Series in Statistics. New York, NY, USA: Springer New York Inc., 2001.

[5] S. B. Kotsiantis, “Supervised Machine Learning: A Review of Classification Techniques,” in *Proceedings of the 2007 Conference on Emerging Artificial Intelligence Applications in Computer Engineering: Real Word AI Systems with Applications in eHealth, HCI, Information Retrieval and Pervasive Technologies*. Amsterdam, The Netherlands, The Netherlands: IOS Press, 2007, pp. 3–24. [Online]. Available: http://dl.acm.org/citation.cfm?id=1566770.1566773

[6] R. Caruana and A. Niculescu-Mizil, "An empirical comparison of supervised learning algorithms," in *Proceedings of the 23rd international conference on Machine learning - ICML '06*. New York, New York, USA: ACM Press, 2006, pp. 161–168. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1143844.1143865

[7] M. Figueiredo, "Adaptive sparseness for supervised learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 9, pp. 1150–1159, sep 2003. [Online]. Available: http://ieeexplore.ieee.org/document/1227989/

[8] C. E. Rasmussen, "Gaussian processes for machine learning," 2006. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.86.3414

[9] C. Harris and M. Stephens, "A Combined Corner and Edge Detector," *Procedings of the Alvey Vision Conference 1988*, pp. 147–151, 1988. [Online]. Available: http://www.bmva.org/bmvc/1988/avc-88-023.html

[10] C. Tomasi, "Good Features," *Image (Rochester, N.Y.)*, pp. 593–600, 1994.

[11] M. S. Extremal, J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust Wide Baseline Stereo from," *In British Machine Vision Conference*, pp. 384–393, 2002.

[12] D. G. Lowe, "Distinctive image features from scale invariant keypoints," *Int'l Journal of Computer Vision*, vol. 60, pp. 91–11 020 042, 2004. [Online]. Available: http://portal.acm.org/citation.cfm?id=996342

[13] K. Mikolajczyk, K. Mikolajczyk, C. Schmid, and C. Schmid, "A performance evaluation of local descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, 2005. [Online]. Available: http://doi.ieeecomputersociety.org/10.110910.1109/TPAMI.2005.188

[14] E. Rosten and T. Drummond, "Fusing points and lines for high performance tracking," *Proceedings of the IEEE International Conference on Computer Vision*, vol. II, pp. 1508–1515, 2005.

[15] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," *Proceedings - 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005*, vol. I, pp. 886–893, 2005.

[16] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 3951 LNCS, pp. 404–417, 2006.

[17] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "BRIEF: Binary robust independent elementary features," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 6314 LNCS, no. PART 4, pp. 778–792, 2010.

[18] S. Leutenegger, M. Chli, and R. Y. Siegwart, "BRISK : Binary Robust Invariant Scalable Keypoints."

[19] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2564–2571, 2011.

[20] A. Alahi, R. Ortiz, and P. Vandergheynst, "FREAK: Fast retina keypoint," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 510–517, 2012.

[21] K. M. Yi, E. Trulls, V. Lepetit, and P. Fua, "LIFT: Learned Invariant Feature Transform," mar 2016. [Online]. Available: http://arxiv.org/abs/1603.09114

[22] H. P. Moravec, "Obstacle avoidance and navigation in the real world by a seeing robot rover." *tech. report CMU-RI-TR-80-03*, p. 175, 1980. [Online]. Available: https://www.ri.cmu.edu/publication{_}view.html?pub{_}id=22

[23] K. Horak, J. Klecka, and P. Novacek, "License plate detection using point of interest detectors and descriptors." Institute of Electrical and Electronics Engineers Inc., nov 2016, pp. 484–488.

[24] S. V. Stehman, "Selecting and interpreting measures of thematic classification accuracy," *Remote Sensing of Environment*, vol. 62, no. 1, pp. 77–89, oct 1997. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0034425797000837

[25] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, 2006. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S016786550500303X

[26] A. Chaturvedi and N. Sethi, "Automatic License Plate Recognition System using SURF Features and RBF Neural Network," in *International Journal of Computer Applications*, vol. 70, no. 27, 2013, pp. 975–8887. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.402.6874{&}rep=rep1{&}type=pdf

[27] F. M. Khaleel, S. N. H. Sheikh Abdullah, and M. K. B. Ismail, "License plate detection based on Speeded Up Robust Features and Bag of Words model," in *2013 IEEE International Conference on Smart Instrumentation, Measurement and Applications (ICSIMA)*. IEEE, nov 2013, pp. 1–5. [Online]. Available: http://ieeexplore.ieee.org/document/6717937/

[28] H. W. Lim and Y. H. Tay, "VEHICLE LICENSE PLATE DETECTION USING UNIGRAM MODEL AND DIFFERENCE-OF-SURF BIGRAM MODEL WITH SVM," 2009. [Online]. Available: http://spict.utar.edu.my/SPICT-09CD/contents/pdf/SPICT09{_}B-1{_}4.pdf