Evaluation of interest point detectors for image information extraction

Iñigo Barandiaran[1,2] Jon Goenetxea[1] John Congote[1,3], Manuel Graña[2], and Oscar Ruiz[3]

[1] Vicomtech
ibarandiaran@vicomtech.org,
[2] Dpto. CCIA, UPV-EHU
[3] EAFIT University

Abstract. Interest points extraction and matching is a common task in many computer vision based application, which are used in different domains, such as 3D reconstruction, object recognition, or tracking. We present an evaluation of current state of the art about interest point extraction algorithms to measure several parameters, such as detection quality, invariance to rotation and scale transformation, and computational efficiency.

Keywords: Computer vision, Interest points, point matching

1 Introduction

Image analysis or computer vision based applications deal with information extraction from the images acquired by a camera sensor. This information is usually discretized in form of relevant pixels or set of pixels (regions) having distinctiveness or discriminative characteristics, i.e, retain information about the structures in the scene being captured, compared with surrounding or neighbouring pixels or regions. Different terms in the literature referring to the same structures can be found, such as feature points, interest points, key points, or corner points. There has been progress in the last decade within the area of feature point extraction algorithms with image and video descriptors that led to substantial improvements in many computer vision areas including registration, 3D reconstruction, motion estimation, image registering, matching,and retrieval, or object and actions recognition. These types of techniques are usually found in the first stages of many computer vision algorithms, and can therefore be considered as low level image information extractors or descriptors. This information is then delivered to other processes in a bottom-up manner, until a semantic knowledge or meaning is recovered. In this article we show the results of an evaluation to measure the behaviour of several points detection algorithms, representative of the current state of the art.

The paper is structured as follows: In section 2 an overview of point detectors followed by a brief description of every detection algorithms evaluated in this study are given. In section 3, a description of the methods, measures and data sets used during the evaluation are shown. In section 4 details about the results obtained during the evaluation are given. Finally, in section 5 some general conclusions are made and future work perspectives are proposed.

2 Methods

2.1 Evaluation Parameters

A review about feature or interest point detectors can be found in [13]. The authors suggest that there are several parameters of a point detector that can be measured, and point out which are the most relevants that a point extractor should address:

Repeatability : Ideally, the same points should be extracted by a detector even if a transformation to the image is applied.

Distinctiveness : Detected features should be different enough in order to be identified or matched, and the same feature should not vary in images of the same scene.

Quantity : This factor measures the number of interest points a detector is able to extract from an image.

Accuracy : Measures the difference in spatial (image) localization of the same point. This value should be as minimal as possible, and ideally having zero or close to zero variance.

Efficiency : Measures how fast an interest point detector is able to process an image.

It's worth noticing that all these factors are somehow application dependent, i.e, every computer vision application or image analysis has different needs. While for applications such as real-time simultaneous location and mapping (SLAM) mainly the quantity, and efficiency factors are critical, for an application such as off-line object recognition, repeatability and distinctiveness are more relevant. In any case, aforementioned factors can be reduced to two: quality and efficiency. Quality represents the feasibility of a detector to delegate accurate, precise, dense and robust set of points to the next process in the image analysis pipeline, while efficiency represents how fast and with which computation resources is able to carry out that task. Depending on these factors, the following processes in the pipeline should or should not apply different mechanisms, filters, estimators, or heuristics in order for the application to succeed, running efficiently and obtaining accurate results. Interest points have to be ideally invariant or robust to any image transformations in order to be identified or matched from one image to another. Image transformations can be categorized in two different classes: geometric and radiometric transformations. Geometric transformations are those that modify the shape or the location of a feature in the image space, while radiometric transformations influence the feature appearance, i.e, the intensity or color value of the pixels. Changes in lighting conditions or camera acquisition parameters, i.e, sensor sensibility, exposure time or lens aperture, directly affect radiometric appearance of image by changing their luminance and/or chrominance information. Several geometric transformations are involved in the process of image formation, starting from the most general ones such as projectivities or homographies, to the most specific ones such as euclidean transformations, rotations, translations and scaling. In our study we focus only in geometric transformations excepting those related to optics such as lens distortions are out of the scope of this study.

2.2 Point Detectors

We have included in our evaluation many of the point detectors that currently represent the state of the art and also some older approaches that today are still broadly used by the computer vision community. In the following section, a brief description of every point detector included in the evaluation is given.

HARRIS We have included Harris corner detector in the evaluation because it's one of the most used feature extractors by computer vision community since its publication [3]. Harris' approach can be seen as an evolution of one of the earliest corner detectors published, such as the Moravec's detector [10]. Harris' approach improves Moravec's detector by taking into consideration different orientations around the candidate pixel, instead of shifting patches by computing the second moment matrix, also called the auto-correlation matrix. Harris cornerness measure is still used by many point extractor approaches as a mechanism of non-maxima suppression. In this evaluation we have used a pyramidal version of Harris proposed in [7].

GFTT (GoodFeatureToTrack) detector [12] is, like Harris detector, one of the most common feature detectors used by computer vision community. This approach was proposed as interest point detector for to be used in camera tracking algorithms, based on previous work of [6]. This approach proposes to different measures named texturedness and dissimilarity for solving the feature matching in order to obtain accurate tracking results, by estimating both a pure translations and affine motion respectively. Texturedness measure is used to extract good candidate points to be tracked, by examining the minimum eigenvalues of auto-correlation matrix.

SIFT (Scale Invariant Feature Transformation) descriptor [5] is one of the most successful approaches for feature or interest point extractor and description. Detection is based on the convolution of images with difference of Gaussians (DoG) operator $\eta = (g_\sigma - g_{\sigma'})$. Images are arranged in a pyramidal representation, where every level(octave) of the pyramid represents a down sampled and smoothed version of the image in previous level. Smoothed images are obtained by convolving with a Gaussian operator with different values of scale σ. This arrange of images allows SIFT to work in a scale-space representation [4] or framework. SIFT detector and respective descriptor is intended to be invariant to either rotation and scale transformation, but not to perspective transformation, such as affine transformation.

SURF (Speed Up Robust Feature) extractor follows an similar approach to SIFT, but addressing the problem of reducing computation cost. SURF searches for local maxima of the Hessian determinant in the scale-space. SURF calculates Hessian determinant efficiently by using a discrete approximation of the Gaussian second order partial derivatives, in conjunction with integral images representation [14]. Differently from SIFT approach, the scale is not obtained by decreasing the image size after smoothing, but by increasing the discrete kernels size.

FAST detector [11] uses a different approach than SIFT or SURF detectors. This approach uses supervised classification to label pixels as their values or pertinence to two classes, interest point or background, by examining the values of surrounding pixels around a candidate one in a circular manner. A feature is detected at pixel p if the intensities of at least n contiguous pixel of a surrounding circle of j pixels are all below or above the intensity of p by some threshold t. FAST approach does not use scale-space representation, and therefore is not invariant to scale nor rotation transformation.

MSER (Maximally stable extremal regions)[9] is an approach based on the detection of blob like structures. MSER detects blobs by using local extrema in intensity or luminance space, obtained by iteratively applying watershed based segmentation. A region R_i is considered a feature if for all its n joined connected components $R_1, ..., R_n$ obtained after n watershed segmentations, it attains to a local minimum in the function $q_i = |R_{i+\alpha} - R_{i-\alpha}|/|R_i|$, where α is a used defined parameter and $|.|$ represents the cardinality of the blob measured in pixels.

STAR This point extractor is also known as Censure (Center Sorround Extrema) [1]. This approach approximates the Laplacian not using DoG operator as SIFT does, but using bi-level center-surround filters of different shapes such as boxes, octagons, or hexagons. The computation of these filters in combination with integral images allows the detection of interest points in scale-space much faster than SIFT. In our evaluations we used bi-level star shaped filter as proposed and implemented in [2].

ORB is the acronym of Oriented BRIEF. This algorithm proposes a modified version of FAST detector for computing orientation during detection step, and an efficient computation of BRIEF based approach for generating descriptors. This approach tries to merge the rotation and scale invariance of SIFT and the computational efficiency of FAST detector.

3 Evaluation

In our evaluation we have used the popular data set, publicly available in [8]. Original data set is composed of 6 different set of images, covering many problems addressed by computer vision or image analysis such as geometric transformations between images, image blurring, changing light conditions or image artifacts due to compression. We used only those set of images covering aspects related with geometric transformations, named Graffiti, Bark and Brick, as shown in figure 1. All tests were carried out using the point detector implementations integrated in Open Source computer vision Library OpenCV[2]. We have focused our evaluation by measuring two key factors in any interest point extractor: repeatability, also known as stability, and accuracy. Stability is referred as the ability of a detector of detect the same feature point across even in a change in radiometric or geometric conditions occurs between two different image acquisitions. Accuracy is related with the consistency and precision, in image space coordinates, in the location of every point extracted by a detector after a change occurs between two images. We have evaluated how accurate every point extractor is by measuring the matching ratio, i.e, the number of matched points or correspondences between two images divided by the total number of points detected, knowing a priori the geometric transformation applied between both images. A candidate pair of points in $images_i$ and $image_j$ respectively is considered a correspondence, if the true match is within a search window of radius r of its estimated correspondence given known rotation, scale or homography transformation. All tests were carried out by using a computer running Windows 7 OS, with 4Gb of RAM and an Intel QuadCore 2.7Ghz CPU. In 2.3.1 version of OpenCV not every CPU point detector algorithm has its own GPU version, so we decided not to distort the results by mixing CPU and GPU implementations.

Fig. 1. (Left) Graffiti Data set. (Center) Bark Data set. (Right) Brick Data set

3.1 Number of points detected

In this test we evaluated how many interest point every extractor is able to detect. Depending of the specificities of every algorithm, the number of extracted features may vary significantly, even if they are apply on the same image. Furthermore, depending on the spatial frequencies present in the images the number of detect points must be different. For example, images taken from Brick data set exhibit high frequency details represented by the bricks on the wall and therefore is the data set that clearly generates the higher number of detections in every extractor.

Table 1. Number of Points detected in Image 1 of every Data Set

Nº of features	Graffiti	Brick	Bark
FAST	**7244**	**41266**	**11735**
STAR	870	1461	168
SIFT	2749	8371	4161
SURF	3203	7142	2253
ORB	702	702	702
MSER	516	2287	837
GFTT	1000	1000	100
HARRIS	855	1000	1000

As shown in table 1 FAST is clearly the one that gets more dense clouds of points, follow by SIFT and SURF, while STAR detector exhibits irregular results. ORB detector seems to have a maximum number of detections allowed, because the same number of 702 features is detected in every image, independently on the content. Similar results are obtained with GFTT and HARRIS having a maximum threshold of 1000 detections. It is important to notice that not only the number of points detected means a successful detector, but also how accurate and repeatable they are against geometric transformations.

3.2 Rotation Transformation

In this test we have evaluated how different approaches are robust against image rotation. We have used image 1 of every data set, and generate artificial images by apply different angle of in-plane rotation starting from 0 (same image), to 57.3 degrees in steps of 11.4 degrees, not varying scale nor perspective transformation. We included the same image without rotation, i.e 0 degrees, as a matter of measure the consistent of the detectors. In this test all detectors but MSER perform similarly, being SIFT, GFTT and ORB the ones that obtain better results.

3.3 Scale Transformation

In the following test we evaluated how the extractors are robust against scaling transformation. We used the image labeled as 1 of every data set, and generated

Table 2. Results of Rotation transformation

Graffiti	0,0	11,4	22,9	34,3	45,8	57,3	*Mean*
2 FAST	100	40,0	41,0	39,3	38,6	40,2	*39,8*
STAR	100	57,3	49,8	50,5	52,9	49,0	*51,8*
SIFT	100	62,4	56,8	56,1	54,3	56,6	*57,2*
SURF	100	44,9	29,2	24,5	24,2	26,2	*29,0*
ORB	100	**82,6**	**82,4**	**78,9**	**76,9**	74,5	***79,0***
MSER	100	33,3	26,7	24,2	20,7	24,2	*25,5*
GFTT	100	79,7	75,5	74,1	71,2	74,2	*74,9*
HARRIS	100	73,4	76,3	75,7	71,7	**75,4**	*74,5*

Bark	0,0	11,4	22,9	34,3	45,8	57,3	*Mean*
FAST	100	52,7	51,4	47,6	46,8	44,6	*56,6*
STAR	100	57,7	44,6	45,8	48,8	39,8	*60,8*
SIFT	100	68,5	61,5	59,1	**57,3**	**55,7**	*65,4*
SURF	100	62,0	45,4	40,6	37,4	35,7	*31,7*
ORB	100	**72,6**	59,4	53,1	49,0	45,2	*66,6*
MSER	100	30,7	22,4	14,0	11,5	11,4	*35,7*
GFTT	100	70,1	**66,3**	**59,1**	55,8	53	*63,8*
HARRIS	100	36,1	38	17	11,1	50,6	***69,0***

Brick	0,0	11,4	22,9	34,3	45,8	57,3	*Mean*
FAST	100	59,6	58,2	56,6	55,0	53,9	*48,5*
STAR	100	70,4	63,3	60,0	56,5	55,0	*47,0*
SIFT	100	69,5	67,3	65,1	63,3	62,0	*60,3*
SURF	100	46,4	31,7	28,6	27,8	27,2	*43,3*
ORB	100	70,2	68,3	66,3	**66,2**	62,3	*55,1*
MSER	100	45,4	37,3	34,0	32,3	31,0	*16,6*
GFTT	100	68,1	63,8	63,7	61,6	61,9	***60,5***
HARRIS	100	**72,5**	**69,6**	**69,6**	66,2	**67,5**	*26,5*

All Images	100	50,1	49,7	47,3	46,3	45,9	47,8
STAR	100	61, 5	52,0	51,8	52,6	47,5	52,9
SIFT	100	66,7	61,7	60,0	58,2	58,0	60,8
SURF	100	50,5	34,8	30,5	30,1	29,4	34,3
ORB	100	74,9	69,4	65,2	62,9	59,4	**66,2**
MSER	100	35,9	28,2	22,6	19,7	20,5	24,7
GFTT	100	72,5	68,4	65,3	62,6	62,4	66,1
HARRIS	100	57,7	58,7	44,7	37,5	63,6	51,5

artificial images by applying different isotropic image scaling factors starting from 1.0 (no scaling) to a factor of 3, not applying rotation nor perspective transformations. As expected, as scale factor increases results of all detectors

Table 3. Results of Scale transformation

Graffiti	1,0	1,4	1,8	2,2	2,6	3,0	*Mean*
FAST	100	37,7	29,8	28,8	29,5	24,3	*29,7*
STAR	100	43,2	31,6	22,0	21,7	13,9	*24,6*
SIFT	100	45,3	27,3	18,6	16,5	12,7	*21,7*
SURF	100	41,5	31,3	28,0	20,1	18,7	*26,7*
ORB	100	**69,2**	51,7	37,7	31,7	**31,0**	*42,1*
MSER	100	31,5	23,8	15,8	8,3	5,4	*13,9*
GFTT	100	67,9	**56,4**	**48,1**	**45,2**	30,5	***47,9***
HARRIS	100	64,6	50,7	44,0	41,0	26,6	*43,5*

Bark	1,0	1,4	1,8	2,2	2,6	3,0	*Mean*
FAST	100	43,2	33,6	30,9	29,8	26,2	*32,3*
STAR	100	50,5	40,4	32,1	24,4	13,6	*29,3*
SIFT	100	46,9	27,5	19,3	13,3	9,6	*19,9*
SURF	100	49,3	34,6	25,2	18,3	12,8	*25,1*
ORB	100	**70,2**	**53,7**	**46,1**	**41,5**	**42,1**	***49,7***
MSER	100	25,0	17,5	10,9	6,3	4,4	*10,5*
GFTT	100	55,5	43,4	36,2	35,3	25,9	*38,0*
HARRIS	100	35,7	23,9	25,3	26,7	15,2	*24,4*

Brick	1,0	1,4	1,8	2,2	2,6	3,0	*Mean*
FAST	100	41,1	33,0	32,6	33,4	27,0	***33,1***
STAR	100	24,9	8,3	2,9	1,5	0,6	*3,5*
SIFT	100	34,0	15,6	8,9	4,8	3,2	*9,3*
SURF	100	37,6	24,7	17,5	11,2	8,6	*17,3*
ORB	100	**53,1**	**33,9**	**26,6**	**24,0**	**20,3**	*29,7*
MSER	100	42,1	29,2	14,2	3,1	0,4	*7,3*
GFTT	100	40,2	27,6	22,7	19,7	18,3	*24,6*
HARRIS	100	43,8	28,6	21,2	21,5	18,8	*26,1*

All Images	1,0	1,4	1,8	2,2	2,6	3,0	*Mean*
FAST	100	40,6	32,1	30,7	31,5	25,9	31,7
STAR	100	37,95	22,0	12,7	9,3	4,8	13,6
SIFT	100	41,6	22,7	14,7	10,2	7,0	16,0
SURF	100	42,5	29,9	23,1	16,0	12,7	22,6
ORB	100	**63,7**	**45,5**	**35,9**	**31,6**	**29,8**	**39,6**
MSER	100	32,1	23,0	13,5	5,5	2,1	10,3
GFTT	100	53,3	40,7	34,1	31,6	24,4	35,5
HARRIS	100	46,6	32,6	28,7	29,9	19,7	30,3

decrease. STAR detector shows the worst performance while GFTT and ORB exhibits the most stable results.

3.4 Homography Transformation

In the following test we have evaluated how well the extractors are robust against homography transformation. An homography is a plane projective transformation. The transformation between any two images of the same planar structure in space can be described by an homography transformation. Homography transformation estimation are widely used by computer vision community in many applications such as image rectification, image registration, camera calibration or camera pose estimation. We have used the six images of Graffiti data set with homography ground truth transformation, mapping points from image 1 to the rest of images of the data set. In this test we obtained overall worst results

Table 4. Results of Homography transformation

Graffiti	Image1	Image2	Image3	Image4	Image5	Image6
FAST	100	25.9	23.4	20.4	16.8	**13.9**
STAR	100	22.2	14.5	6.4	4.8	1.6
SIFT	100	30.8	14.3	9.4	5.8	3.7
SURF	100	20.0	11.8	7.3	5.3	3.9
ORB	100	**54.1**	**36.7**	**27.6**	**17.3**	3.2
MSER	100	16.2	12.7	6.9	5.2	1.7
GFTT	100	33.4	21.5	13.3	10.7	1.9
HARRIS	100	39.4	25.1	14.5	13.6	1.7

compared to rotation or scale tests. The worst results are obtained comparing image1 with image6 (column 6) because of the severe distortion generated by the homography to patches around each feature point. These results shows that no one of the tested extractors are truly invariant to perspective transformation. Most of the current approaches propose to generate feature point detectors and descriptors to be invariant to affine geometric transformation. Affinities preserves parallelism between lines, while projectivities only preserves straightness of lines not giving enough discriminative or distinctiveness power allowing to have a detector fully invariant to such type of transformations. However, perspective transformations can be effectively approximated by piecewise local affinities so affine invariants detectors should perform best in this test. ORB detector clearly outperforms the rest of detectors, being the one that gets best matching ratios in all images of all data sets.

3.5 Accuracy

We wanted also to measure how precise and consistent every point extractor is related with the locations, in image space coordinates, of every point detected, after a viewpoint change occurs between two images. We measure the accuracy by calculating the difference in pixels between location of detected point, being a correct match, and the true location. We depict here the averaged results obtained with Graffiti, brick and bark data sets. Results depicted in table 5 shows that there is no clear winner with respect to feature point location accuracy, given a mean pixel error value of 0,62 between all detectors.

Table 5. Mean pixel error for Homography transformation

Graffiti	image1	image2	image3	image4	image5	image6
FAST	0,00	0,73	0,73	0,74	0,75	0,76
STAR	0,00	0,70	0,72	0,72	0,73	0,77
SIFT	0,01	**0,66**	**0,68**	0,75	0,75	0,81
SURF	0,05	0,73	0,74	0,74	0,77	0,79
ORB	0,25	0,70	0,73	0,73	0,75	0,75
MSER	0,10	0,72	0,70	0,74	**0,70**	**0,74**
GFTT	0,00	0,69	0,70	0,77	0,77	0,84
HARRIS	0,00	0,69	0,71	**0,71**	0,73	0,81

3.6 Efficiency

In addition to accuracy of feature point detectors and their invariance to geometric transformation, how fast and efficient they perform such task is also critical in some computer vision applications. we have measured the time they expend to perform detection task operation in every of the six images of Graffiti data set. As mentioned previously, depending on the content of the images, i.e low or high frequencies, the number of detected points vary and therefore computation time. The lower computation time the better. Results in image 3.6 shows

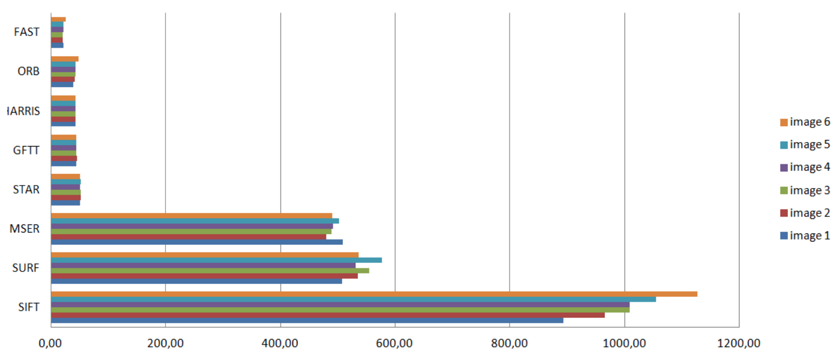

Fig. 2. Computation Time measured in miliseconds (lower time is better)

clearly that FAST is the best performer related with efficiency. Even if being the one that extracts the highest number of points, it only takes 22ms per image on average. It worth notice also how ORB is very close to FAST taking only 42.7 ms per frame, and taking into account that this approach estimates feature orientation while FAST does not. As expected, SIFT extractor is the slower due to the computation of several DoG operators, taking 1000ms on average.

4 Conclusions

We present an evaluation of different feature or interest point extractors. We have measured their invariance and robustness to several geometric image transformations such as rotations, translations, scaling or perspective transformations such as homographies. We have also evaluated their capability of generating information by measuring the number of points they generate. Finally, we also measured their efficiency related with computational resources. As mentioned before, the choice of the feature detector very much depends on the application specifications. Overall, we can conclude that the recent ORB detector obtains the best ratio between accuracy and efficiency. ORB shows the best performance on rotation and homography tests and also being the second related with computational cost only exceeded by FAST, however this one does not have on orientation component and does not produce multi-scale features and therefore is not as accurate as ORB in rotation and scaling transformations. One very important aspect nowadays is efficiency, as more and more applications are being migrated to mobile devices, such as iPad or iPhone. In this way, those approaches similar to FAST or ORB detector requiring low computation and memory resources are very useful and promising. The next step is to evaluate all these algorithms running on a mobile device, taking into account that some implementation may be optimized for running on a specific processor architecture using specific instructions, not supported by mobile processors. We are now interested in evaluating how current feature descriptors, in addition with random sampling robust estimation strategies, can overcome problems of robustness and invariance to scale, rotation, and perspective transformations. Recent feature descriptor approaches such as BRIEF, GLOH, DAISY, or ORB opens new possibilities for computer vision applications, such as simultaneous location and mapping, or image registering and reconstruction to run robustly in real-time.

References

1. M. Agrawal, K. Konolige, and M. Blas. Censure: Center surround extremas for realtime feature detection and matching. *Computer Vision–ECCV 2008*, pages 102–115, 2008.
2. G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.
3. C. Harris and M. Stephens. A combined corner and edge detector. In *Alvey vision conference*, volume 15, page 50. Manchester, UK, 1988.
4. T. Lindeberg. *Scale-space theory in computer vision.* Springer, 1993.
5. D.G. Lowe. Object recognition from local scale-invariant features. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 2, pages 1150–1157. Ieee, 1999.
6. B.D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of the 7th international joint conference on Artificial intelligence*, 1981.
7. K. Mikolajczyk and C. Schmid. Indexing based on scale invariant interest points. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 1, pages 525–531. IEEE, 2001.
8. K. Mikolajczyk, T. Tuytelaars, C. Schmid, et al. Affine covariant features. *Collaborative work between: the Visual Geometry Group, Katholieke Universiteit Leuven, Inria Rhone-Alpes and the Center for Machine Perception*, 2007.

9. K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L.V. Gool. A comparison of affine region detectors. *International journal of computer vision*, 65(1):43–72, 2005.
10. H.P. Moravec. Towards automatic visual obstacle avoidance. In *IJCAI*, 1977.
11. E. Rosten, R. Porter, and T. Drummond. Faster and better: A machine learning approach to corner detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(1):105–119, 2010.
12. J. Shi and C. Tomasi. Good features to track. In *Computer Vision and Pattern Recognition, 1994. Proceedings CVPR'94., 1994 IEEE Computer Society Conference on*, pages 593–600. IEEE, 1994.
13. T. Tuytelaars and K. Mikolajczyk. Local invariant feature detectors: a survey. *Foundations and Trends® in Computer Graphics and Vision*, 3(3):177–280, 2008.
14. P. Viola and M.J. Jones. Robust real-time face detection. *International journal of computer vision*, 57(2):137–154, 2004.