A Systematic Approach for the Parameterisation of the Kernel-Based Hough Transform Using a Human-Generated Ground Truth

Jonas Lang(✉), Mark Becke, and Thomas Schlegl

Regensburg Robotics Research Unit, Faculty of Mechanical Engineering,
Ostbayerische Technische Hochschule Regensburg, Regensburg, Germany
{jonas2.lang,mark.becke,thomas.schlegl}@oth-regensburg.de

Abstract. Lines are one of the basic features that are used to characterise the content of an image and to detect objects. Unlike edges or segmented blobs, lines are not only an accumulation of certain feature pixels but can also be described in an easy and exact mathematical way. Besides a lot of different detection methods, the Hough transform has gained much attention in recent years. With increasing processing power and continuous development, computer vision algorithms get more powerful with respect to speed, robustness and accuracy. But there still arise problems when searching for the best parameters for an algorithm or when characterising and evaluating the results of feature detection tasks. It is often difficult to estimate the accuracy of an algorithm and the influences of the parameter selection. Highly interdependent parameters and preprocessing steps continually lead to only hardly comprehensible results. Therefore, instead of pure trial and error and subjective ratings, a systematic assessment with a hard, numerical evaluation criterion is suggested. The paper at hand deals with the latter ones by using a human-generated ground truth to approach the problem. Thereby, the accuracy of the surveyed Kernel-based Hough transform algorithm was improved by a factor of three. These results are used for the tracking of cylindrical markers and to reconstruct their spatial arrangement for a biomedical research application.

Keywords: Feature detection · Human-generated ground truth · Hough transform · Image processing · Line detection · Systematic parameterisation

1 Introduction

There is still a challenge in automatically detecting shapes for object recognition. It remains a computationally expensive and non-trivial task, even for simple geometric patterns like lines, circles or ellipses. Where humans use additional heuristics, experiences from the past and a deeper image understanding, computers are restricted to more basic algorithms. Noise, missing or extraneous and imperfect data from the initial camera image, preprocessing steps or edge

H. Liu et al. (Eds.): ICIRA 2015, Part I, LNAI 9244, pp. 473–486, 2015.
DOI: 10.1007/978-3-319-22879-2_44

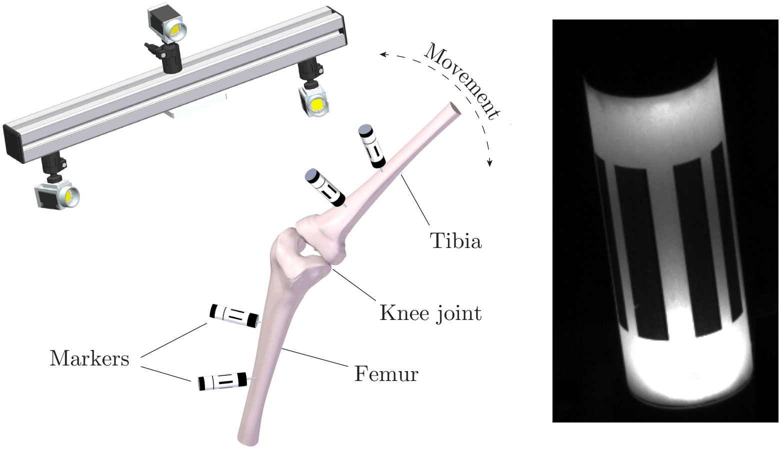


Fig. 1. Application of the marker tracking system and one of the corresponding marker images

detection further complicates the problem. Nevertheless, a large number of simple as well as sophisticated algorithms have been developed. One of the most famous line detection methods during the last decades is the Hough transform (HT). This algorithm transforms a curve searching problem into a much simpler peak searching problem. The effort for further development mainly focuses on computation time, robustness or accuracy. But these advancements often lack the numerical evaluation of results and the investigation of parameter influences. Some in-house or database images and the corresponding, extracted features are shown and sometimes only optically assessed.

To solve computer vision tasks, a whole chain of algorithms and processing steps is commonly used. Individual elements cannot be evaluated separately and without considering their interdependence. However, it is often difficult to examine this processing chain with all its parameters and influences in its entirety. The same is true for a hard, numerical key figure or classification coefficient to rate the end results of the image processing task. To solve this problem, a known ground-truth is required as a basis of comparison for the subsequent results. There are various possibilities, like simulation, measurement of the real scene or human-generated comparison data. All of them have their own strengths and weaknesses and it is highly dependent on the case of application which one to choose.

In the present paper, line and corner detection algorithms are used to reconstruct the spatial arrangement of cylindrical markers. These markers are tracked by a trinocular camera system which is pointed towards the examination object, in our case the bones of an experimental setup for biomechanical testing of knee joints moved manually or automatically, e.g. by a six axis robot, see Fig. 1. Those experiments try to imitate the human gait cycle by moving along a certain path or trajectory with the insertion of a specific force and torque [1]. By these means, long term stress tests can be simulated. In addition to the robot's internal position sensors, a computer vision system monitors the movements of the test object and thereby provides a correction factor for the robot's control loop.

Concerning line detection, a very recent survey analyses more than 200 papers dealing with the Hough transform over the last decades [2]. There is a myriad of derivatives and further developments with the Standard Hough Transform (SHT) as a starting point, like the Fast Hough Transform (FHT) [3], the Randomized Hough Transform (RHT) [4], the Probabilistic Hough Transform (PHT) [5] or the Kernel-based Hough Transform (KHT) [6], just to name a few. The latter will gain particular attention in this paper as it is the investigation subject in Sect. 3. Apart from the Hough transform to detect straight lines, there are some additional algorithms analysing eigenvalues or linking pixel clusters [7–9]. A test framework to assess the accuracy of the line detection process by the Hough transform is proposed in [10]. A local Hough transform to detect line candidates and further determination of their parameters by a global estimation are combined in [11].

The Hough transform generally only detects infinite lines. Nevertheless, there are several algorithms which overcome this restriction and provide information about the length and position or the start and end point of a line, respectively. This can either be the analysis of the neigbourhood of peaks in the parameter space or the application of additional independent processing steps in the image space. Each feature pixel is mapped into the parameter space as a sinusoid curve. A line as a succession of collinear pixels leads to a split up sinusoidal construct, also called butterfly. Depending on the location and length of the line, this butterfly takes a distinctly shaped appearance, see Fig. 2). Using these characteristics and suitable algorithms to interpret it, a finite line can be described completely [12–14].

Many computer vision applications for feature detection lack the existence of absolute, numerically exact comparison data or the real solution or dataset (i.e. true values) respectively. Especially edge detection or segmentation algorithms are often only assessed by a mere subjective verdict. Whereas a human-generated ground truth provides the possibility to get hard, mathematical classification numbers when evaluating the performance of an algorithm with its specific parameter set. Perhaps the most famous example for the use of human-generated ground truth is the *Segmentation Database of Berkley's Computer Vision Group* [15], which is frequently applied to evaluate and compare different segmentation algorithms [16,17]. An extension towards 3D segmentation tasks is realised in [18] and [19], while [20] compares 3D interest point detection algorithms with human selections. Edge and boundary detections were assessed in comparison with human-generated data in [21] whereas [22] and [23] try to introduce a more formal, mathematical comparison model.

The paper at hand is using human-generated comparison data to evaluate the influence of preprocessing steps and parameter choice on line detection through the Kernel-based Hough Transform (KHT) introduced by Fernandes [6]. In contrast to other works, the whole process is considered. The underlying

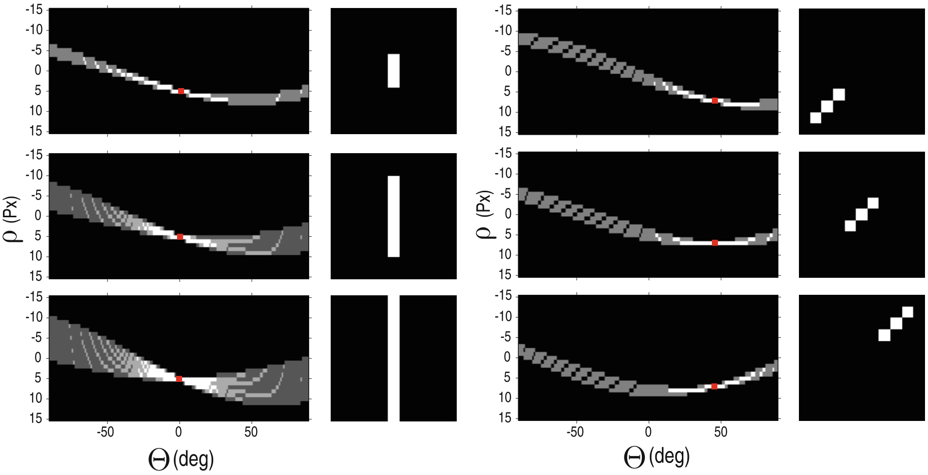


Fig. 2. The corresponding Hough space for different line lengths and positions

image processing sequence is described in Subsect. 2.1, a methodology for a paramaeter influence study using human-generated ground truth is proposed in Subsect. 2.2, mathematical quality factors are introduced in Subsect. 2.3 and exemplary results are given in Sect. 3.

2 Methodology

Because of the highly interdependent parameters of a image processing task, it is almost impossible to find a good or even the best parameter set by just guessing or trying out different combinations non-systematically. Apart from this, as described in Sect. 1, it is problematic to obtain numerical comparison data to assess the results of an algorithm. Both problems are addressed in the subsequent chapters. To solve the first, a systematic parameter study is conducted, to tackle the second, a mathematically describable ground-truth is developed.

2.1 Image Processing Sequence

A flowchart of the image processing chain used for this work and parts of its corresponding results are shown in Fig. 3. Images are acquired from monochrome industrial area cameras through Camera Link connection and PCI-E framegrabbers. In these initial images, the markers only occupy a small fraction of the total image area. Therefore, a region of interest (ROI) creation by using binary thresholding and a blob analysis is necessary to limit the subsequent computational effort. Furthermore, the ROIs are undistorted by the use of calibration data from a previous camera calibration process according to [24]. Afterwards, the contrast is adjusted, noise is reduced by Gaussian filtering and edges are

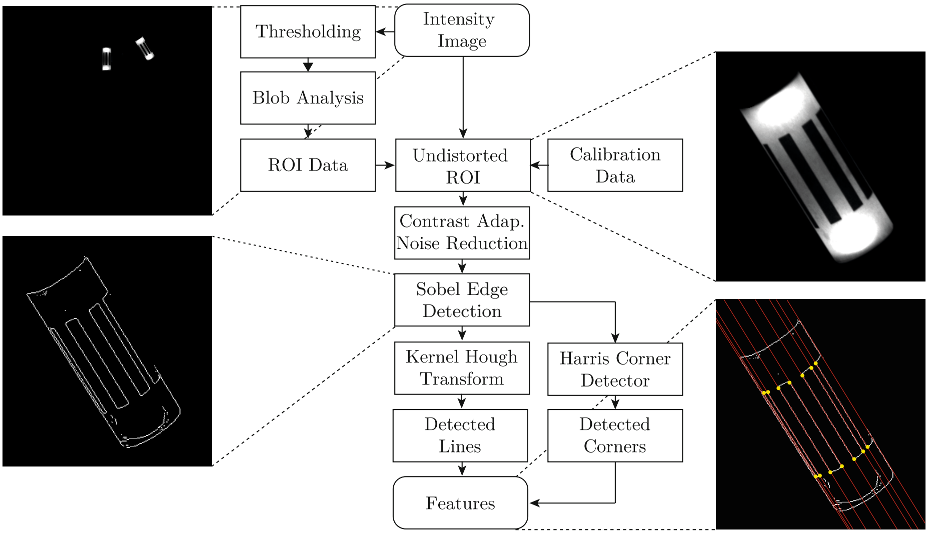


Fig. 3. Flowchart of the proposed image processing algorithm and the corresponding results

detected by the use of Sobel or Canny algorithms. The whole image processing chain is realised in Matlab. For time-efficiency several *OpenCV Library* functions are utilised.

To detect lines, the Standard Hough transform individually transforms each binary edge feature pixel into the parameter space. In contrast to that, the KHT links connecting feature pixels, subdivides them into collinear clusters and votes for these with a Gaussian voting scheme (see Fig. 4 for details). This leads to several advantages in comparison to the SHT. The computational effort is, depending on the image, reduced by a factor up to ten [6] and the parameter space is much clearer, with solitude real peaks and less noisy sub-peaks. As a result, the robustness and accuracy are increased and false positive line detection candidates are reduced. For most HT implementations, lines are used in their Hesse normal form, i.e., the line for a fixed ρ and Θ is

$$\rho = u \cos \Theta + v \sin \Theta \, \forall \, u, v \in \mathbb{R}. \tag{1}$$

Instead of evaluating the neighbourhood of peaks as described in Sect. 1, the results of a Harris corner detector are utilised to determine the end points of a line. This is less critical with respect to nearby lines or peaks and uses the corner characteristics of the rectangular areas on the marker design, see Fig. 3. The Harris feature points provide further possibilities to validate and optimise (i.e., error minimise) the reconstruction results. All the corner points at the top and at the lower end lie on an elliptical curve each. This property creates additional evaluation and cross checking data. The same is true for one upper and one lower (i.e., the start and end point) Harris feature for one line. Those two

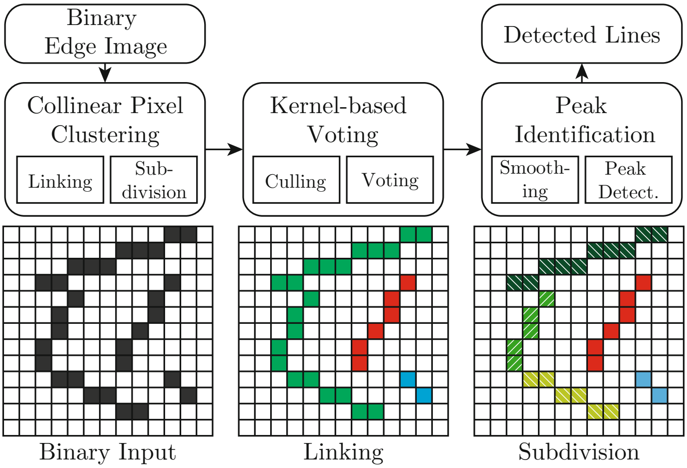


Fig. 4. Flowchart of the KHT algorithm with a exemplary linking and subdivision process

points span an additional line with certain θ and ρ values that can be compared to the corresponding θ and ρ values of the KHT line.

2.2 Human-Generated Ground Truth

There are various possibilities to achieve a hard, numerical ground-truth as a comparison base. First of all, a computational simulation and mapping of the real scene can be utilised. However, it is hard to take all aspects into consideration with influencing factors like discretization of pixel space, optical distortion or marker lighting and the ambient light situation. Second, a measurement of the exact position and rotation of both, all markers and all cameras, can be conducted. But this approach is very time-consuming and only feasible with special measurement equipment like a coordinate measurement machine or a measurement arm. Additionally, the procedure is always error-prone. Finally, a human-generated ground truth can be used. For this goal, different test subjects manually mark all the lines in all the images. Certainly, there is no exact true location for a line. It is a partly subjective task to tag the start and end point or the profile of a line when there is no hard black to white contrast but a grayscale gradient. Thus, averaging for each line marked by all persons leads to favourable results. This takes into consideration that every human being has a different perceptual view and minimizes those differences. Furthermore, outliers (e.g., wrongly marked lines by accident) are eliminated.

2.3 Mathematical Description

Several options for an evaluation criterion when comparing the distance or correspondence of two lines are possible. As the line data are available in their normal form, see (1), the most basic approach would be the distance in Hough space or the geometrical distance of two point pairs ρ and Θ for both lines, respectively. However, a difference of one pixel ρ would lead to the same distance rating result as one degree Θ, which does not represent the true circumstances. Therefore, the distance of the perpendicular foot (d_p) of two lines and the angular difference of its normal vectors (d_a) (see Fig. 5) are used. Depending on the position of the lines in the image or on the sensor, the latter leads to slightly different results. This is caused by the fact that when the perpendicular foot is displaced by one pixel, the influence on d_a is higher, the further away it lies from the principal point. Both will be used subsequently, the first for its descriptiveness with easy to grasp pixel units in image space, the second for its exact nature in the real three-dimensional scene. Alternatively to d_a the cosine distance of the two vectors $\boldsymbol{v}_{\mathrm{l1}}$ and $\boldsymbol{v}_{\mathrm{l2}}$ can be used, which leads to a reduced computational effort.

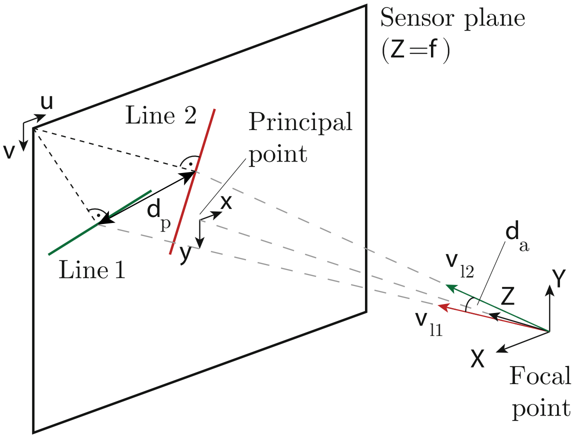


Fig. 5. Sensor plane with two lines and their corresponding perpendicular foot distance (dp) and perpendicular foot angle (da)

As an evaluation benchmark, the results of the KHT with its related parameter and preprocessing sets are compared to the human-generated ground truth. Therefore, all lines are allocated to each other and their distances are calculated. To assess the parameter set, the mean accuracy value of all lines in all 60 images is computed. Parameter sets which lead to incomplete detection results (i.e., in at least one image, at least one line has not been detected) are rejected. The 20 most significant lines (i.e., those with the highest bin or peak in the parameter space) are allocated and compared to the, depending on the view of the marker, eight to ten human-generated master lines.

The indices l_1 and l_2 represent the human-generated master line and its corresponding auto-detected KHT line which should be compared. n_{img} is the total number of all evaluated images, n_{line} the number of lines in each image. The perpendicular foot distance can be computed by

$$d_{\mathrm{p}} = \sqrt{(u_{\mathrm{l}_1} - u_{\mathrm{l}_2})^2 + (v_{\mathrm{l}_1} - v_{\mathrm{l}_2})^2} \tag{2}$$

on the image plane with

$$u = \rho \cos \Theta \tag{3}$$

and

$$v = \rho \sin \Theta \tag{4}$$

and its mean value with

$$\bar{d}_{\mathrm{p}} = \frac{1}{n_{\mathrm{line}} n_{\mathrm{img}}} \sum_{i=1}^{n_{\mathrm{img}}} \sum_{j=1}^{n_{\mathrm{line}}} d_{\mathrm{p}}(i,j). \tag{5}$$

In addition, the mean angular difference of the normal vectors with $\boldsymbol{v}$ as the vector pointing from the principal point towards the perpendicular foot of a line (see Fig. 5) can be calculated by

$$d_{\mathrm{a}} = \arccos \frac{\boldsymbol{v}_{\mathrm{l}_1} \cdot \boldsymbol{v}_{\mathrm{l}_2}}{\|\boldsymbol{v}_{\mathrm{l}_1}\| \, \|\boldsymbol{v}_{\mathrm{l}_2}\|} \tag{6}$$

in the three dimensional space using

$$\bar{d}_{\mathrm{a}} = \frac{1}{n_{\mathrm{line}} n_{\mathrm{img}}} \sum_{i=1}^{n_{\mathrm{img}}} \sum_{j=1}^{n_{\mathrm{line}}} d_{\mathrm{a}}(i,j) \tag{7}$$

together with

$$\boldsymbol{v} = \begin{pmatrix} u_0 - \rho \cos \Theta \\ v_0 - \rho \sin \Theta \\ f \end{pmatrix} \tag{8}$$

where u_0 and v_0 are the coordinates of the principal point and f the focal length of the camera. Those parameters are individually evaluated for each camera using intrinsic camera calibration with a standard chequered pattern.

3 Parameter Study Results

The marking procedure for the realisation of the human-generated ground truth was implemented as a MATLAB script. The users always had the possibility to zoom in and out during the whole marking process to assess details of the line just as the whole line profile at once. The intensity images and not the already binarised edge images are used to generate the ground-truth, thus taking into account the influence of the edge detection algorithm and threshold, too. The data basis for all evaluations are 20 spatial poses of a cylindrical marker seen from three cameras which leads to a total of 60 images. Fifteen people have participated in the marking procedure. The task at hand was explained by a short demonstration and illustrated help files. Fig. 6 shows a flowchart for the parameter study process.

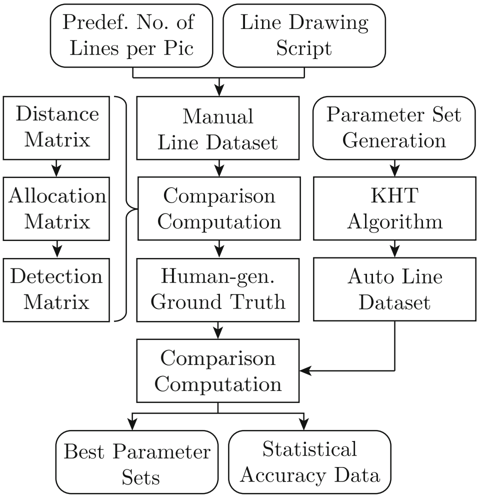
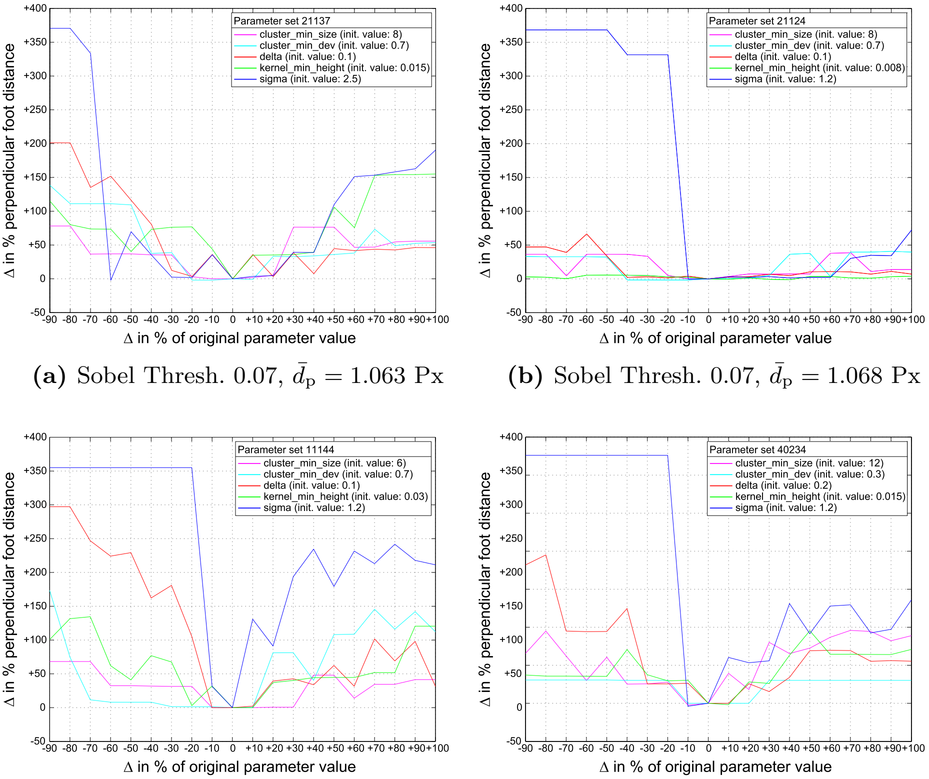


Fig. 6. Flowchart of the KHT parameter study

All of the five KHT parameters (see Table 1 for details) are varied in 10 steps and arranged in all possible combinations, leading to a total of $1 \cdot 10^5$ permutations. Additionally, the preprocessing is distinguished in six different alternatives: Sobel and Canny edge detection with three thresholds each. For the Sobel algorithm a Gaussian filter with the same parameters as the one for the Canny algorithm was applied. Although both algorithms actually use the same filter mask, there is a difference in thresholding and edge thinning. Therefore, both lead to different results. The standard MATLAB implementation of both algorithms with the thresholds 0.07, 0.10 and 0.13 is used.

The best five results of the parameter study for each edge detection algorithm and threshold combination can be seen in Table 3. Preferable results are achieved with Sobel edge detection and a relatively low threshold. In general, the Sobel results are about 50 percent more accurate than the Canny ones. In comparison to the original default parameter values (cf. Table 1) with a resulting $\overline{d_{\mathrm{p}}}$ of 3.326 Pixels, the best parameter set 21137 is more than three times more accurate ($\overline{d_{\mathrm{p}}} = 1.062$ Pixels) and almost in a sub-pixel range. Using the angle d_{a} as a distance criterion instead of the perpendicular foot distance d_{p}, the order of the best parameter sets stays the same. Table 2 shows a comparison for both variations. Their ratio varies only slightly depending on the relative position of a line in the image. Towards the border of the image sensor, one pixel difference leads to a larger angular difference then towards the image center (i.e., close to the principal point). A comparison between the detected default lines and the improved lines is shown in Fig. 8.

Figure 7 shows the effects on the detection accuracy when a single parameter is varied around its initial value. Parameter set 21137 is a narrow choice (see Fig. 7a), where small variations of the parameters lead to a rapid decrease of



(a) Sobel Thresh. 0.07, $\bar{d}_{\mathrm{p}} = 1.063$ Px

(b) Sobel Thresh. 0.07, $\bar{d}_{\mathrm{p}} = 1.068$ Px

(c) Sobel Thresh. 0.10, $\bar{d}_p = 1{,}174$ Px

(d) Sobel Thresh. 0.13, $\bar{d}_p = 1{,}099$ Px

Fig. 7. Sensitivity of the detection accuracy to variations of single parameters

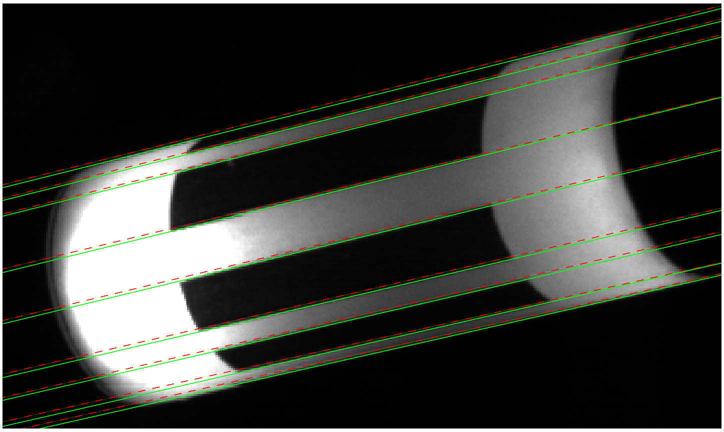


Fig. 8. Comparison between lines with the default parameter set (red, dashed) and lines with the improved parameter set (green, solid)

Table 1. Overview of the KHT parameters, its default values and the range for the parameter study

Parameter	Symbol	Def. value	Exam. range	Description
Cluster_Min_Size	c_σ	10	[2, ..., 25]	Minimal size of a cluster to be still considered as such
Cluster_Min_Dev	c_δ	2	[0.3,...,5]	Minimal distance for a feature pixel to be assigned to a cluster
Delta	δ	0.5	[0.03,...,1.5]	Discretisation of the parameter space
Kernel_Min_Height	κ	0.002	[0.001,...,0.4]	Minimum height of a kernel to still pass culling
Sigma	σ	2	[0.3,...,5]	Standard deviation for the Gaussian kernel

Table 2. Comparison between *d*p and *d*a as a distance criterion with Sobel edge detection (Threshold = 0.07)

$\bar{d}_\mathrm{p}$ (Px)	SD(d_p) (Px)	$\bar{d}_\mathrm{a}$ (°)	SD(d_a) (°)
1.062	0.274	$2.016 \cdot 10^{-2}$	$0.521 \cdot 10^{-2}$
1.067	0.293	$2.026 \cdot 10^{-2}$	$0.556 \cdot 10^{-2}$
1.076	0.301	$2.043 \cdot 10^{-2}$	$0.571 \cdot 10^{-2}$
1.078	0.309	$2.046 \cdot 10^{-2}$	$0.586 \cdot 10^{-2}$
1.081	0.309	$2.052 \cdot 10^{-2}$	$0.587 \cdot 10^{-2}$

the accuracy (steep response curve), whereas parameter set 21124 (see Fig. 7b) lies in a much broader area with favourable parameters (flat response curve, except σ), where parameter changes lead to only marginally worse results. The same can be seen in Table 3d, where, in contrast to row one, the rows two to five represent similar parameter choices. The lowering of $\sigma = 2.5$ in parameter set 21137 (Fig. 7a) of 60 percent almost equals the $\sigma = 1.2$ of parameter set 21124 (Fig. 7b). Variations and especially a lowering of σ generally lead to the steepest rise in inaccuracy. Smaller δ values, i.e. a finer rasterisation of the parameter space, mainly lead to worse results. Bins of a certain size are necessary to avoid oversampling with too many sub peaks. Small σ values cause that peaks of different cluster do not melt into combined peaks anymore, which leads to worse results because of unintended sub-peaks of separate clusters only representing a part of the actual, entire marker line.

Table 3. Best five parameter sets and their corresponding perpendicular foot values with statistical values in (Px)

(a) Canny edge detection (Threshold 0.07)

$\bar{d}_{\mathrm{p}}$	SD(d_{p})	Min(d_{p})	Max(d_{p})	KHT Parameter Vector
1.475	0.481	0.863	2.232	12124: (6 1.1 0.10 0.008 1.2)
1.476	0.485	0.863	2.265	11124: (6 0.7 0.10 0.008 1.2)
1.479	0.455	0.915	2.222	1114: (4 0.7 0.10 0.004 1.2)
1.481	0.465	0.915	2.289	1104: (4 0.7 0.10 0.002 1.2)
1.483	0.491	0.928	2.248	1004: (4 0.7 0.05 0.002 1.2)

(b) Canny edge detection (Threshold 0.10)

$\bar{d}_{\mathrm{p}}$	SD(d_{p})	Min(d_{p})	Max(d_{p})	KHT Parameter Vector
1.467	0.476	0.877	2.208	11124: (6 0.7 0.10 0.008 1.2)
1.473	0.495	0.877	2.301	21124: (8 0.7 0.10 0.008 1.2)
1.474	0.483	0.883	2.222	1124: (4 0.7 0.10 0.008 1.2)
1.476	0.489	0.916	2.261	1004: (4 0.7 0.05 0.002 1.2)
1.480	0.495	0.877	2.301	20124: (8 0.3 0.10 0.008 1.2)

(c) Canny edge detection (Threshold 0.13)

$bard_{\mathrm{p}}$	SD(d_{p})	Min(d_{p})	Max(d_{p})	KHT Parameter Vector
1.494	0.475	0.915	2.359	1114: (4 0.7 0.10 0.004 1.2)
1.505	0.488	0.900	2.313	11105: (6 0.7 0.10 0.002 1.6)
1.509	0.498	0.915	2.359	11114: (6 0.7 0.10 0.004 1.2)
1.510	0.500	0.892	2.359	10114: (6 0.3 0.10 0.004 1.2)
1.514	0.440	0.889	2.169	11216: (6 0.7 0.20 0.004 2.0)

(d) Sobel edge detection (Threshold 0.07)

$\bar{d}_{\mathrm{p}}$	SD(d_{p})	Min(d_{p})	Max(d_{p})	KHT Parameter Vector
1.062	0.274	0.672	1.506	21137: (8 0.7 0.1 0.015 2.5)
1.067	0.293	0.772	1.619	21124: (8 0.7 0.1 0.008 1.2)
1.076	0.301	0.741	1.598	21125: (8 0.7 0.1 0.008 1.6)
1.078	0.309	0.730	1.598	21134: (8 0.7 0.1 0.015 1.2)
1.081	0.309	0.710	1.695	21136: (8 0.7 0.1 0.015 2.0)

(e) Sobel edge detection (Threshold 0.10)

$\bar{d}_{\mathrm{p}}$	SD(d_{p})	Min(d_{p})	Max(d_{p})	KHT Parameter Vector
1.173	0.336	0.435	1.526	40234: (12 0.3 0.20 0.015 1.2)
1.219	0.334	0.895	1.830	2004: (4 1.1 0.05 0.002 1.2)
1.225	0.414	0.464	1.969	31224: (10 0.7 0.20 0.008 1.2)
1.232	0.344	0.896	1.829	13004: (6 1.5 0.05 0.002 1.2)
1.241	0.379	0.867	1.859	23004: (8 1.5 0.05 0.002 1.2)

(f) Sobel edge detection (Threshold 0.13)

$\bar{d}_{\mathrm{p}}$	SD(d_{p})	Min(d_{p})	Max(d_{p})	KHT Parameter Vector
1.098	0.385	0.456	1.706	11144: (6 0.7 0.1 0.03 1.2)
1.105	0.404	0.469	1.768	21144: (8 0.7 0.1 0.03 1.2)
1.187	0.561	0.513	2.521	10144: (6 0.3 0.1 0.03 1.2)
1.242	0.380	0.717	1.882	32244: (10 1.1 0.2 0.03 1.2)
1.252	0.648	0.466	2.804	31144: (10 0.7 0.1 0.03 1.2)

4 Conclusion

An image processing chain to detect features for the calculation of the spatial arrangement of cylindrical markers has been presented. To improve its results, a parameter study with the use of a human-generated ground truth was conducted and statistically evaluated. This increased the accuracy of line detection results by a factor of three. Using this data, the movements of objects during the biomedical testing of knee prosthesis with a six axes robot can be assessed and used as a correction factor for the robot's control loop.

The proposed method can be adapted to other computer vision problems where complex parameter sets lead to manifold influencing factors which are only hard to survey. Instead of a merely subjective verdict, numerical comparison data can be achieved.

As future work, the results will be further assessed and cross checked with simulation and a measurement ground-truth. Furthermore, the implication of feature detection accuracy on the precision of the reconstruction algorithm will be evaluated.

References

1. Becke, M., Schlegl, T.: Toward an experimental method for evaluation of biomechanical joint behavior under high variable load conditions. In: IEEE International Conference on Robotics and Automation (ICRA), pp. 3370–3375 (2011)
2. Mukhopadhyay, P., Chaudhuri, B.B.: A Survey of Hough Transform. Pattern Recognition **48**(3), 993–1010 (2015)
3. Li, H., Lavin, M.A., Master, R.J.L.: Fast Hough Transform: A Hierarchical Approach. Computer Vision, Graphics, and Image Processing **36**(2–3), 136–161 (1986)
4. Li, Q., Xie, Y.: Randomised Hough Transform With Error Propagation for Line and Circle Detection. Pattern Analysis and Applications **6**(1), 55–64 (2003)
5. Kiryati, N., Eldar, Y., Bruckstein, A.: A Probabilistic Hough Transform. Pattern Recognition **24**(4), 303–316 (1991)
6. Fernandes, L.A.F., Oliveira, M.M.: Real-time Line Detection Through an Improved Hough Transform Voting Scheme. Pattern Recognition **41**(1), 299–314 (2008)
7. Akinlar, C., Topal C.: Real-time line segment detection by edge drawing. In: 18th IEEE International Conference on Image Processing (ICIP), pp. 2837–2840, September 2011
8. Montero A., Nayak A., Stojmenovic M., Zaguia N.: Robust line extraction based on repeated segment directions on image contours. In: Computational Intelligence for Security and Defense Applications (CSIDA), pp. 1–7 (2009)
9. von Gioi, R., Jakubowicz, J., Morel, J.-M., Randall, G.: LSD: A Fast Line Segment Detector with a False Detection Control. IEEE Transactions on Pattern Analysis and Machine Intelligence **32**(4), 722–732 (2010)
10. Nguyen, T.T., Pham X.D., Kim, D., Jeon, J.W.: A test framework for the accuracy of line detection by hough transforms. In: 6th IEEE International Conference on Industrial Informatics (INDIN), pp. 1528–1533 (2008)
11. Guerreiro, R., Aguiar, P.: Incremental local hough transform for line segment extraction. In: 18th IEEE International Conference on Image Processing (ICIP), pp. 2841–2844 (2011)

12. Furukawa, Y., Shinagawa, Y.: Accurate and Robust Line Segment Extraction by Analyzing Distribution Around Peaks in Hough Space. Computer Vision and Image Understanding **92**(1), 1–25 (2003)
13. Du, S., Tu, C., van Wyk, B., Chen, Z.: Collinear segment detection using HT neighborhoods. IEEE Transactions on Image Processing, 3612–3620 (2011)
14. Dai, B., Pan, Y., Liu, H., Shi, D., Sun, S.: An Improved RHT algorithm to detect line segments. In: 2010 International Conference on Image Analysis and Signal Processing (IASP), pp. 407–410 (2010)
15. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A Database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Proc. 8th International Conference on Computer Vision, pp. 416–423 (2001)
16. Martin, D., Fowlkes, C., Malik, J.: Learning to detect natural image boundaries using local brightness, color, and texture cues. IEEE Transactions on Pattern Analysis and Machine Intelligence, 530–549 (2004)
17. Crevier, D.: Image Segmentation Algorithm Development Using Ground Truth Image Data Sets. Computer Vision and Image Understanding **112**(2), 143–159 (2008)
18. Attene, M., Katz, S., Mortara, M., Patane, G., Spagnuolo, M., Tal, A.: Mesh segmentation - a comparative study. In: IEEE International Conference on Shape Modeling and Applications (2006)
19. Benhabiles, H., Vandeborre, J.-P., Lavoue, G., Daoudi, M.: A Framework for the objective evaluation of segmentation algorithms using a ground-truth of human segmented 3D-models. In: IEEE International Conference on Shape Modeling and Applications (SMI), pp. 36–43 (2009)
20. Dutagaci, H., Cheung, C.P., Godil, A.: Evaluation of 3D Interest Point Detection Techniques via Human-generated Ground Truth. The Visual Computer **28**(9), 901–917 (2012)
21. Wang, S., Ge, F., Liu, T.: Evaluating edge detection through boundary detection. EURASIP Journal on Advances in Signal Processing, 213–227 (2006)
22. Lopez-Molina, C., De Baets, B., Bustince, H.: Quantitative Error Measures for Edge Detection. Pattern Recognition **46**(4), 1125–1139 (2013)
23. Yitzhaky, Y., Peli, E.: A Method for Objective Edge Detection Evaluation and Detector Parameter Selection. IEEE Transactions on Pattern Analysis and Machine Intelligence **25**(8), 1027–1033 (2003)
24. Zhang, Z.: A Flexible New Technique for Camera Calibration. IEEE Transactions on Machine Intelligence **22**(11), 1330–1334 (2000)