Deep Learning Based Automated Vickers Hardness Measurement

Ehsaneddin jalilian[1][0000−0001−5925−5735] and Andreas Uhl[1][0000−0002−5921−8755]

[1] Department of Computer Science, University of Salzburg
Jakob-Haringerstraße 2, Salzburg, Austria
{ejalilian,uhl}@cs.sbg.ac.at

Abstract. Automated Vickers hardness measurement remains to be a challenging task due to the difficulties associated with Vickers indentation detection and complex specimen surface defects. Typical image processing methods fail to detect the indentations in specimens possessing rough and noisy surfaces, distorted indentation shapes, or cracks. We propose a robust deep learning based model for accurate automated Vickers hardness measurement in this work. A Fully Convolutional Neural network (FCN) is chosen to accurately localize and segment the Vickers indentations. A set of liner curves are then fitted to the boundary pixels data extracted from the output segmentations. The initial positions of the indentation vertices are estimated as the cross-sectional point of adjacent boundaries to each indentation vertex. A complimentary segmentation module then is used to refine the target regions, and accurate indentation vertices positions are then calculated applying further geometric processing steps. The accuracy of the model is compared to known algorithms from the literature and results are presented. The evaluation is conducted on two significant indentation image databases with 150 and 216 highly varying images.

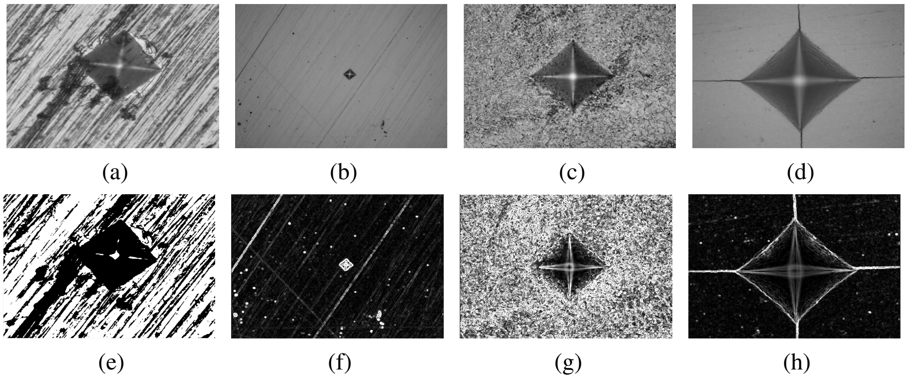
Keywords: Vickers Hardness Measurement · Deep Learning.

1 Introduction and Related Work

Vickers hardness testing is a technique used for examining the resistance of the surface of solid specimens such as metals, ceramics, or polymers. In this method hardness testing machines called durometers apply a pyramidal diamond indenter with an angle of 136° between opposite faces on the specimen surface, with measurable load and time period. The Vickers hardness (VH) is the quotient obtained by dividing the applied force load (F) by the square area of indentation:

$$VH = \frac{1}{g} \frac{2Fsin\frac{136^\circ}{2}}{d^2}, \tag{1}$$

where d is the mean of the diagonal length of the indentation, and g is the acceleration of gravity (see Fig. 2 for an illustration). Manual inspection and measurement of indentations is time consuming and very interpretive [11]. Computer-assisted hardness testing systems have been developed to address these issues and also to provide more accurate measurements. Accurate detection and localization (segmentation) of the indentations in the Vickers images is the most crucial step for correct measurement of the indentation dimensions. In fact, the indentations vary significantly in terms of: size, location, rotation, brightness, contrast and texture characteristics in the corresponding images. On the other hand, the target specimens normally have noisy surfaces which contain cracks, sparkles and other industrial distortions. The computational methods proposed so far for automatic indentation detection and measurement fail to perform well in certain groups of images depending on the techniques applied. Thresholding algorithms [11] [10] easily get misled by the global distribution of the background (specimen surface) pixels which contribute extensively to the threshold level computation (see Figs. 1a and 1e for an example). Long overtures, speckles, and break lines that cross through edges or that emit from vertices may disturb the reconstruction algorithms as used in the edge detection approaches [8] [2]. Figs. 1b, 1c and 1d show example images and their corresponding outputs (Figs. 1f, 1g and 1h) using a Sobel edge operator. Also, lack of significant markers, or rough specimen surface, or industrial defects may cause the Axes projection algorithms [11] to fail to detect the significant differences (which correspond to the indentation region) in such images. Template matching methods [2] [3] are computationally expensive and their matching mechanism is very sensitive to the orientation of the indentation, structural discrepancies and artifacts on the specimen surface. Active-contour algorithms [4] generally require parameters that are distribution dependent and very hard to generalize. Benefiting from the recent advancements in deep learning technology, Tanaka *et al.* [12] used two CNN modules separately to detect the bounding-box surrounding the Vickers indentation, and the left indentation vertex positions. In fact the feasibility of this model is very restricted as two seprate training sessions, and thus two training datasets along with their corresponding manual ground-truths are required to initialize the system. The authors further used physical deviation (in terms of μm) measure (instead of pixel-wise measurement as used in other related research) to evaluate the model performance, and thus it is difficult to validate their results against other commonly used algorithms.


(a) (b) (c) (d)

(e) (f) (g) (h)

Fig. 1: Sample indention images (first row), and their output segmentations using adaptive threshold (1e, 1f), and Sobel edge detection algorithm (1g, 1h)

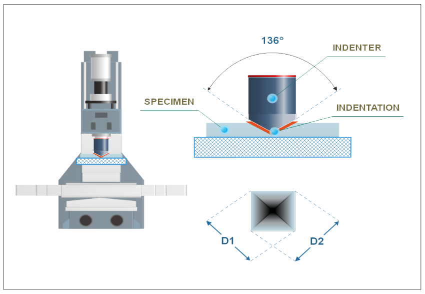


Fig. 2: Durometers and Vickers indention hardness test schematics

This study aims to leverage the power of deep learning networks within a simple and viable framework to propose a robust and simplified automated Vickers indentation measurement model. In particular, a powerful semantic segmentation Fully Convolutional Neural Network (FCN) with strong emphasis on boundary pixel preservation is chosen to perform the challenging task of Vickers indentation region detection and segmentation. The Vickers indentation boundary information obtained in this stage then is used as the basis to fit (four) linear curves to the indentation edges. The initial Vickers indentation vertices positions are approximated, at the sub-pixel level, computing the cross-sectional points of the adjacent curves. A Region Of Interest window (ROI) is defined around each vertex position, and the target regions are further refined to extract the actual indention vertices using an adaptive segmentation module. The vertices final positions then are calculated utilizing a geometry-based technique. The robustness and accuracy of the model is verified on samples that differ substantially in terms of indention shape, size, type, and distribution of the noises.

2 Methodology

As we already mentioned, the key challenging step in automatic Vickers indentation testing is localization and segmentation of the indentation in the images. So, we selected to handle this task utilizing a segmentation CNN, as the models proved to provide superior performance in wide variety of application, specially in complicated and challenging segmentation tasks [6] [7].

2.1 Indentation Segmentation using Convolutional Neural Network

As a key criteria, the network needed to possess very strong profile in preserving boundary pixels data, as later the detected boundary data is used as the input for estimating the initial indention vertices' positions. We selected the RefineNet [9] to accomplish this task, as the network is already proven to enable high-resolution prediction, and at the same time to preserve the boundary information. We used an ADAM optimizer with learning rate of 0.0001, executing 40,000 iterations to train the network [1].

[1] https://github.com/eragonruan/refinenet-image-segmentation

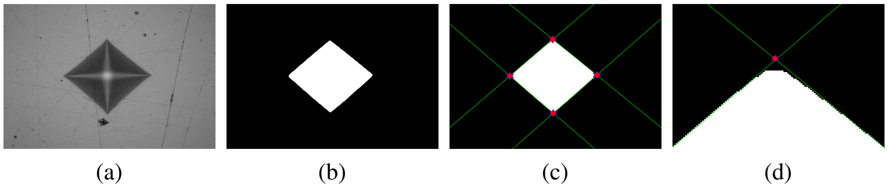


Fig. 3: A sample input image (3a), its corresponding output segmentation (3b), the curves fitted to the indention four edge points (3c), and an initial vertex position specified (3d)

2.2 Edge Extraction and Initial Indention Vertex Position Estimation

The FCN outputs provide the input data to extract the indentation edge points information. In this way, we first refine the output segmentations by extracting the biggest segmentation profile (by size), thereby removing the false-positive outliers, in the output segmentations. All connected objects in the output masks are found, and their sizes are calculated (pixel-wise) to find the biggest one. Then, the indentation contour points are extracted [5] and a first degree polynomial curve is fitted into each indentation boundary point utilizing a least-squares criterion. The actual indentions' shapes are not fully squared and the edges have rather concave profile, which bend towards the inside as we move along the edges from vertex points toward the middle of the edges. To this extent, we considered an estimate (50%) of uncertainty values (based on the standard deviation of the edge points) to be added to the estimated curve positions. This compensated for the nonlinearity property of the edges and found (by experiment) to improve the algorithm precision notably. The intersection point of the crossing curves L_1 and L_2 with the line segments defined as $(x_1\ y_1), (x_2\ y_2)$ and $(x_3\ y_3), (x_4\ y_4)$ then are calculated to determine the initial coordinates of the indentation vertices $(V_{x,y})$:

$$\begin{aligned} V_x &= \frac{(x_1y_2 - y_1x_2)(x_3 - x_4) - (x_1 - x_2)(x_3y_4 - y_3x_4)}{(x_1 - x_2)(y_3 - y_4) - (y_1 - y_2)(x_3 - x_4)} \\ V_y &= \frac{(x_1y_2 - y_1x_2)(y_3 - y_4) - (y_1 - y_2)(x_3y_4 - y_3x_4)}{(x_1 - x_2)(y_3 - y_4) - (y_1 - y_2)(x_3 - x_4)}. \end{aligned} \tag{2}$$

Fig. 3 demonstrates a sample input image (3a), its corresponding output segmentation (3b), the curves fitted (3c), and an initial vertex estimated (3d).

2.3 Precision Improvement

To this extent, we leveraged the segmentation power of the FCN to get the challenging task of indentation positioning and segmentation accomplished. We also estimated (or rather predicted) the initial indentation vertices positions based on the information obtained in this stage. However, the actual vertex information did not contribute much to these initial estimations. This was mainly due to the fact that corner region pixels went missing in the output segmentations (see Fig. 3d). Missing corner pixels is a general segmentation issue and is not specific to the CNN-based segmentation models. So, in

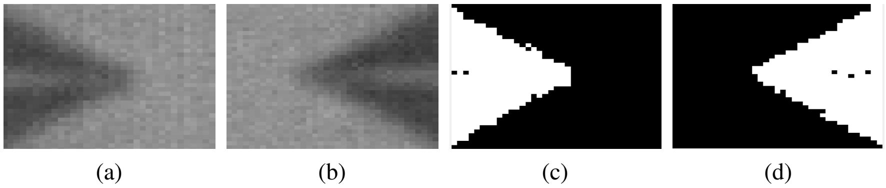


Fig. 4: A Sample right (4a) and left (4b) ROI images and their corresponding output segmentations (4c, 4a), using the complementary segmentation module

the next step we considered to extract the actual missing vertex pixels, and further process this information to improve the initial vertices position estimations. For this, first we defined a region of interest (ROI) window with size 40×40 around each initial vertex position. The typical background (specimen surface) distortions already mentioned are minimized or even not existing in such limited region, and vertex features are fairly differentiable from the background pixels. Close inspection of the ROI images (Figs. 4a, 4b) and their outputs segmentations (Figs. 4c, 4d) reflected this fact properly too. Therefore, we chose to utilize Otsu's adaptive clustering algorithm to segment the ROI region into the foreground (corresponding to the indentation corner pixels) and the background (corresponding to the specimen surface pixels). To perform the clustering, the algorithm maximizes inter-class (w) variance (which is equivalent to to minimizing the intra-class variance):

$$\sigma_w^2(t) = \omega_0(t)\omega_1(t)\left[\mu_0(t) - \mu_1(t)\right]^2 . \tag{3}$$

The class probability $\omega_0(t)$ and $\omega_1(t)$ are computed from the L bins of the histogram:

$$\omega_0(t) = \sum_{i=0}^{t-1} p(i) \ , \quad \omega_1(t) = \sum_{i=t}^{L-1} p(i), \tag{4}$$

and the class means $\mu_0(t)$, and $\mu_1(t)$, are calculated as:

$$\mu_0^2(t) = \sum_{i=1}^{t} \left[i - \mu_1(t)\right]^2 \frac{P(i)}{w_1(t)'} \ , \ \ \mu_1^2(t) = \sum_{i=t+1}^{I} \left[i - \mu_2(t)\right]^2 \frac{P(i)}{w_2(t)}. \tag{5}$$

As it can be seen in the output results (Figs. 4c, 4d), the vertex regions are effectively separated from the background. The straight-forward approach to specify the vertex

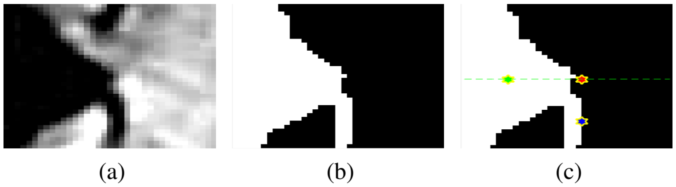


Fig. 5: A sample ROI with connected defect (5a) and its corresponding output segmentation (5b), where the initial vertex point (blue mark) is corrected (red mark), considering the vertex central gravity (green mark) in the output (5c)

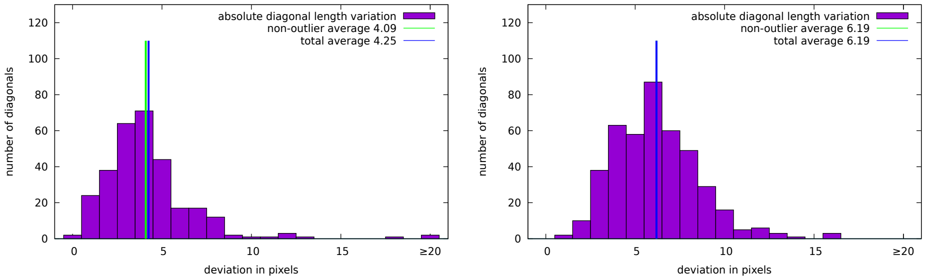


Fig. 6: Histogram of the maximum deviation for the manual measurements in DA (left) and DB (right) respectively

position in the output segmentations is to find the coordinate of the most extreme pixels located at the tip of the segmented indentation corner region. However, locating such a pixel ought not to be a trivial task. In some cases the defects in the surface (grooves or holes) are connected to the indentation corner regions (see an example in Fig. 5). While the target vertex pixel is assumed to be a single pixel located at the tip of the segmented area (see Fig. 4c), yet it turned out in certain cases to have linear shape (formed from two or more pixels) in some. To address these issues, we decided to calculate the horizontal coordinate of the vertex point based on the ”central gravity” of the segmented area. So, we calculated the centroid $(C_{x,y})$ of the segmented area [1] as follows:

$$\begin{aligned} C_{\mathrm{x}} &= \frac{1}{6A}\sum_{i=0}^{n-1}(x_i + x_{i+1})(x_i y_{i+1} - x_{i+1} y_i) \\ C_y &= \frac{1}{6A}\sum_{i=0}^{n-1}(y_i + y_{i+1})(x_i y_{i+1} - x_{i+1} y_i)\,, \end{aligned} \tag{6}$$

Where the area (A) of the polygon containing the segmented area is defined as:

$$A = \frac{1}{2}\sum_{i=0}^{n-1}(x_i y_{i+1} - x_{i+1} y_i)\,. \tag{7}$$

We also calculated the vertical coordinate of the vertex point, averaging the top-left and top-right pixels’ vertical coordinates of the polygon containing the segmented area. As illustrated in Fig. 5, this approach enables us to effectively filter out the connecting outliers affect, and at the same time allows us to compute the actual coordinates of the vertex in case of vertices having a linear shape.

3 Experimental Framework

We used two Vickers hardness indentation databases to carry out our experimental studies which has been used in previous work [2, 3]. Each database contains a substantial number of images (DA: 150, DB: 216) at a resolution of 1280 × 1024 pixels, which were captured directly in production operation environments. Images contain one Vickers micro-indentation with severe variation in the size, location and rotation of the indentation, the texture of the specimen surface, and the overall focus and contrast of the picture. We utilized an annotation tool to generate the manual ground-truths of indentation images required to train the FCN network. The indentation regions were marked (in the corresponding binary ground-truth mask), fitting a square-shaped polygon (manually) to each indention region. We requested multiple testing experts to generate the diagonal measurement manual ground-truths, and the vertices manual ground-truths. Fig. 6 shows a histogram of the maximum deviation for the manual measurements for the databases DA and DB respectively, which can be considered as a human baseline for the desired quality of automatic hardness measurement algorithms.

4 Experiments and Results

We selected representatives of indentation detection algorithms from Section 1, as well as the proposed model as described in Section 2 to be evaluated against the two indentation image databases described in Section 3. To evaluate our model, first we trained the FCN network on the images in the databases. We applied a 2-fold cross evaluation training scheme for the network. For this, we partitioned each database into two equal parts, and then trained and tested the network alternatively on each database partition. Doing so, we tested the networks on all samples in each database without overlapping the training and testing sets. After obtaining the output segmentations for each database, we applied the measurement pipeline as explained in Section 2 to the output segmentations. Results were compared to the median of the manual measurements and presented in terms of two averages errors: the overall average errors (reflecting the algorithm robustness and accuracy), and the non-outlier average error in which the errors exceeding 20 pixels are inhibited (reflecting the algorithm precision). Likewise, we evaluated the other algorithm representatives on the databases as well (i.e. we have taken their accuracy results from [3]). Table 1 lists the algorithms together with the results obtained in these experiments. The axes projection algorithm evaluated here differs from the original proposal [11], as Otsu's algorithm is used to determine the threshold level for the binarization step as well as to determine the threshold level for the indentation detection in the x- and y-projection. Furthermore, all images were rotated by 45° based on the requirements of the algorithm.

Considering the primary measurement errors (non-outlier average diagonal errors), the proposed model shows superior performance in comparison with the other algorithms on both databases (2.43 pixels for DA and 1.51 pixels for DB). The results

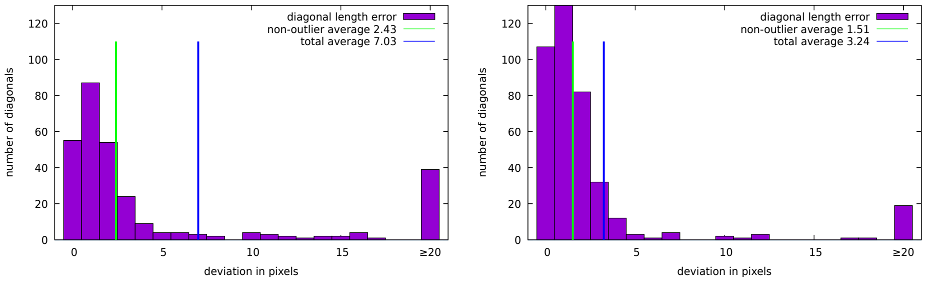


Fig. 7: Deviation of the diagonal lengths from the ground-truths for the database DA (left) and DB (right) respectively

Table 1: Diagonal indentation measurement results for different algorithms along with the corresponding manual measurements

Measure	Average Diagonal Error (pixels)		Non-outlier Average Diagonal Error (pixels)	
Database	DA	DB	DA	DB
Axes projection [11]	62.01	48.04	8.49	8.18
Vertex template matching [2]	10.01	4.65	4.13	4.24
Edge template matching [2]	14.16	5.37	4.18	4.96
Edge tracing [2]	13.73	5.61	4.15	4.97
Ray sweeping [2]	24.58	14.70	12.03	12.09
Three-stage template matching [3]	8.75	4.77	2.96	2.90
Ours	7.03	3.24	2.43	1.51
Manual measurements	3.12	4.28	3.06	4.30

proved to be very precise on both databases as they are even better than the average manual measurements. Furthermore, considering the average diagonal measurements errors (7.03 pixels for DA and 3.24 pixels for DB), the model delivers a robust performance as well, and proves to deal better with hard cases compared to the other algorithms (see Fig. 7 for the corresponding error graphs). Among the other algorithms representatives, the three-stage template matching algorithm shows better results (in terms of both measurement errors) compared to the others. The axes projection and ray sweeping algorithms result in a vast number of outliers, and generally do not deliver promising results. The vertex and the edge template algorithms however perform better (than the two previous algorithms), stil worse compared to the proposed one.

In addition to the evaluation of the accuracy of diagonal length measurements, we also examined the positional error of the indentation vertices for the proposed model. Examination of the vertex errors is not necessary for the sole computation of the hardness value of a specimen but has significance in imitating the manual hardness measuring process where measurement lines are attached to the indentation in the image. Such measurement lines are likewise welcome in automated measurements to enable verification and monitoring of the operation. Minimal positional errors of the indentation vertices are therefore demanded to ensure good visual feedback. Fig. 8 shows the histogram of the deviation of vertex positions from the ground-truths for both databases,

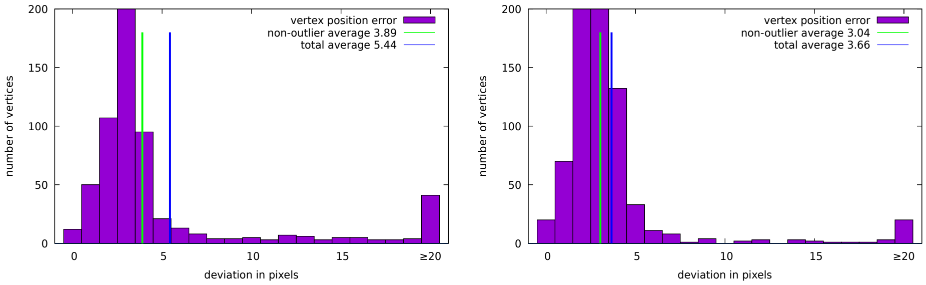


Fig. 8: Deviation of vertex positions from the ground-truths for the database DA (left) and DB (right) respectively

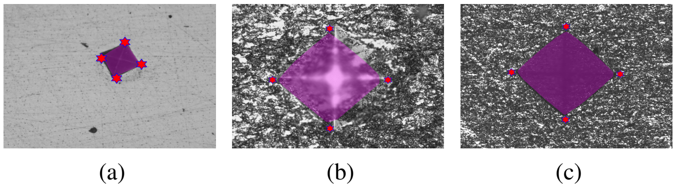


Fig. 9: Examples of problematic cases where the model does not yield good results

where the non-outlier average error is 3.89 and 3.04 pixels for the DA and DB databases, respectively. The deviation is computed according to the procedure described in Section 3. Fig. 9 shows examples of problematic cases where the model does not yield promising results. For both problem cases, the first step of the model, which is the indentation localization and segmentation, failed and generated false indentation boundaries that misled all the following steps. In particular, Fig. 9a shows a segmentation defect (underestimation) on both the upper-right and the lower-right indentation boundaries which placed the indentation vertices' initial positions quite far from the actual vertices positions, as well as the ROI windows. This of course is due to the FCN segmentation error, and needed to be analyzed from this point of view. The significance of Figs. 9b and 9c are however quite questionable, because the indentations completely lack contrast to the surrounding. This violates the structural features learned by the FCN, and thus the indention region is just guessed by the network here.

5 Conclusion

The proposed deep learning based model showed superb results for automatic Vickers indentation measurement in the Vickers images. The FCN network enabled us to successfully perform the challenging task of indentation localization and segmentation, specially in hard cases where indentation profiles are distorted by rough specimen surface, sparkles, and low contrasts. The segmentation network delivered reliable input data to the secondary segmentation module where indentation vertices positions were extracted. The accuracy of the predicted vertex positions were further improved applying a geometric improvement technique, and thereby the model delivered the results that were superior to all studied competing algorithms in terms of both the exactness of the measured diagonal lengths and the robustness. The size of the databases enabled the calculation of statistics, which give quantitative predictions about the viability of an algorithm when deployed. It is also shown that the precision of the proposed algorithm is even better than manual measurements performed by operators and thus lays beyond the scope of human measurement variations. The accuracy of the model was highly influenced by the network segmentation performance. So, enhancing the segmentation accuracy specially in the indentation corner regions will diffidently mitigate the segmentation related errors as those discussed in Section 4. The key competences of the proposed algorithm are its robustness to the size, location and rotation of the indentation in the images as well as to the brightness conditions of the images, and the resistance against surface defects. Moreover, simplicity of the techniques applied, and absence of tunable parameters, makes it a practical module for Vickers indention measurement.

References

1. Bourke, P.: Calculating the area and centroid of a polygon. Swinburne Univ. of Technology **7** (1988)
2. Gadermayr, M., Maier, A., Uhl, A.: Algorithms for microindentation measurement in automated vickers hardness testing. In: Tenth International Conference on Quality Control by Artificial Vision. vol. 8000, p. 80000M (2011)
3. Gadermayr, M., Maier, A., Uhl, A.: Robust algorithm for automated microindentation measurement in vickers hardness testing. Journal of Electronic Imaging **21**(2), 021109 (2012)
4. Gadermayr, M., Maier, A., Uhl, A.: Active contours methods with respect to vickers indentations. Machine Vision and Applications **24**(6), 1183–1196 (2013)
5. Hert, S., Schirra, S.: 2d convex hulls and extreme points. In: CGAL User and Reference Manual. CGAL Editorial Board, 5.2.1 edn. (2021)
6. Jalilian, E., Uhl, A.: Iris segmentation using fully convolutional encoder–decoder networks. In: Bhanu, B., Kumar, A. (eds.) Deep Learning for Biometrics, chap. 6, pp. 133–155. Springer, (ZG) Switzerland (2017)
7. Jalilian, E., Uhl, A.: Finger-vein recognition using deep fully convolutional neural semantic segmentation networks: The impact of training data. In: Proceedings of the IEEE 10th International Workshop on Information Forensics and Security. pp. 1–8. Hong Kong (2018)
8. Jiand, Y., Xu, A.: A new method for automatically measurement of vickers hardness using thick line hough transform and least square method. In: International Congress on Image and Signal Processing. pp. 1–4. IEEE (2009)
9. Lin, G., Milan, A., Shen, C., Reid, I.: Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1925–1934 (2017)
10. M-Nicolas-Dominguez, S., Wiederhold, P.: Indentation image analysis for vickers hardness testing. In: International Conference on Electrical Engineering, Computing Science and Automatic Control. pp. 1–6. IEEE (2018)
11. Takao, S., Tadao, K.: Development of an automatic vickers hardness testing system using image processing technology. IEEE transactions on Industrial Electronics **44**(5), 696–702 (1997)
12. Tanaka, Y., Seino, Y., Hattori, K.: Automated vickers hardness measurement using convolutional neural networks. The International Journal of Advanced Manufacturing Technology **109**(5), 1345–1355 (2020)