SmartKC++: Improving the Performance of Smartphone-Based Corneal Topographers

Vaibhav Ganatra[1] Siddhartha Gairola[2] Pallavi Joshi[3] Anand Balasubramaniam[3]
Kaushik Murali[3] Arivunithi Varadharajan[4] Bellamkonda Mallikarjuna[4] Nipun Kwatra[1]
Mohit Jain[1]

[1] Microsoft Research, India [2] Max Planck Institute of Informatics, Saarbrücken, Germany
[3] Sankara Eye Hospital, India [4] Remidio Innovative Solutions Private Limited, Bengaluru, India

Abstract

Keratoconus, an ocular condition marked by progressive corneal thinning and outward bulging, presents diagnostic challenges due to the high cost and lack of portability in conventional corneal topographers. These limitations restrict accessibility for many, necessitating affordable and mobile alternatives. Innovations like SmartKC [9] offer a low-cost and portable alternative, however, there still remains some gaps in performance when compared to commercial topographers. In this paper, we introduce SmartKC++, a series of innovative methodological improvements to the image processing pipeline of SmartKC, aimed at significantly enhancing its diagnostic precision and reliability. Our comprehensive evaluation on a dataset comprising 303 eye images reveals that SmartKC++ boosts the accuracy of automated keratoconus diagnosis by 7.69% relative to SmartKC.

1. Introduction

Keratoconus is an ocular disorder in which the cornea—the clear, dome-shaped front of the eye—gradually bulges outward into an irregular conical shape. The prevalence of keratoconus is varied, ranging from 0.5% in the US [14] to 2.3% in India [13]. Early treatment options for keratoconus include corrective glasses and collagen cross-linking surgery, whereas severe cases require corneal transplants. Due to the high surgery cost and limited availability of corneas for transplant, keratoconus is one of the leading causes of partial and complete blindness in middle- and low-income countries [11].

The clinical gold standard for diagnosing keratoconus involves estimating the curvature of corneal surface using a technique called corneal topography. This procedure, performed using medical devices such as Optikon Keratron and Oculus Pentacam, is non-invasive. However, these devices are bulky, non-portable, costly (Pentacam costs upwards of USD 20,000), and require skilled technicians to operate. Hence, they are not accessible to a large portion of the population. To address this, several smartphone-based screening methods have been proposed [2, 9, 19]. Many of these methods involve attachments to capture images of the eye, which are then processed for keratoconus detection. Among them, SmartKC[1] [9] stands out as the only technique that outputs tangential and axial heatmaps of corneal curvature, along with diagnostic metrics such as sim-K values [6]. It achieves this by adapting the operating principle of corneal topographers to smartphones. The key components of SmartKC are: (1) a 3D-printed placido-disc attachment to project concentric rings onto the cornea, (2) a data-collection smartphone app for reliable data capture (incorporating quality checks for tilt and offset), and (3) an image processing pipeline that analyzes reflected placido-disc patterns (called *mires*) to estimate corneal curvature by reconstructing the corneal surface. For more details on the workings of SmartKC, please refer [9].

Although a promising low cost and portable alternative, SmartKC does exhibit some limitations. For example, in the original SmartKC evaluation [9], 11.4% of the collected data was discarded due to the presence of "broken mires", despite the quality checks implemented in the data collection app. The wide variability in the quality of input images may arise due to the hand-held nature of data capture, tear film artifacts in the eye, etc. These broken mires lead to errors in downstream mire segmentation and localization, highlighting a key limitation in the robustness of the SmartKC's image processing pipeline. Also, the mean absolute error in the sim-K1 values reported by SmartKC and medical corneal topographers ranges from 4.22 (in the dis-

[1] https://github.com/microsoft/SmartKC-A-Smartphone-based-Corneal-Topographer

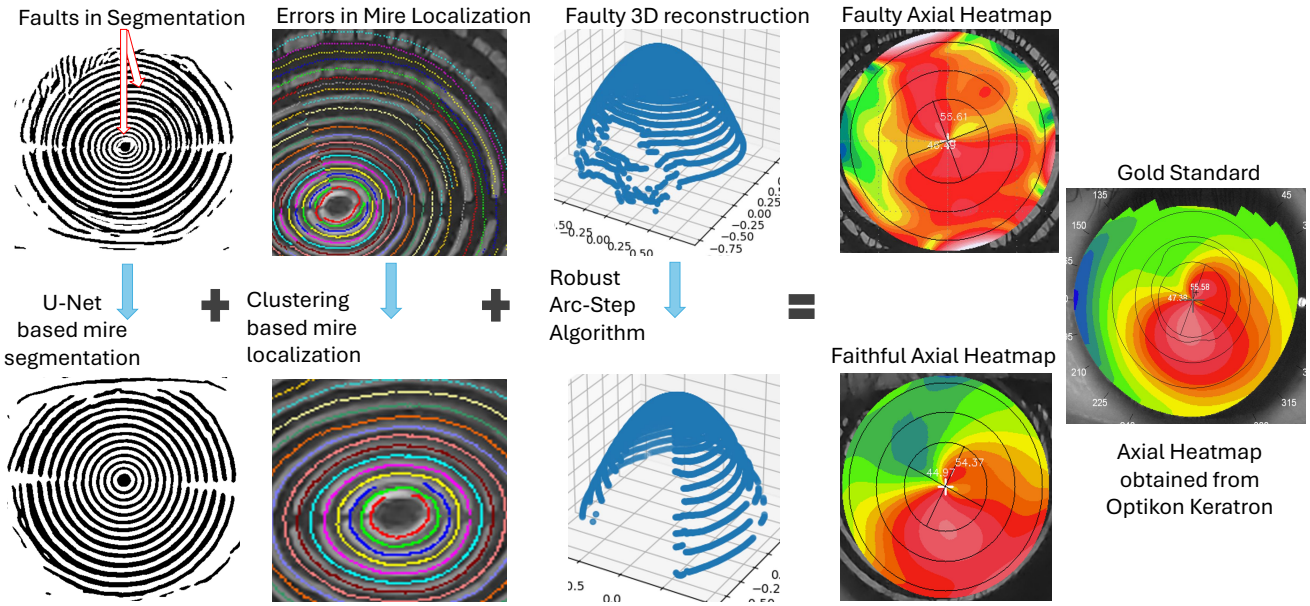

Figure 1. Overview of *SmartKC++* - changes made to the mire segmentation, localization and corneal surface reconstruction mechanisms to enhance the performance and robustness of the system

carded images) to 1.29 (in the non-discarded images), underscoring the need for improvement.

To address these limitations, we propose *SmartKC++*, comprising of three novel modifications to the SmartKC image processing pipeline (Figure 1): (1) improved *mire segmentation* using a U-Net model trained on labels generated from traditional image processing techniques, (2) improved *mire localization* using a clustering algorithm that utilizes local spatial information in the image to identify broken mires, and (3) improved *Arc-Step method* for corneal surface reconstruction to accommodate any missing points due to broken mires.

We evaluate *SmartKC++* on a real-world dataset of 303 eye images collected from 163 patients. Our findings demonstrate a significant improvement in system robustness, leading to a reduction in the discard rate of collected images from 11.4% to 0%. Our algorithm automatically addresses broken mires, resulting in reliable heatmaps. We observe qualitative enhancements in the generated heatmaps (Figure 2) and quantitative improvements in the agreement of the sim-K values. The mean absolute error in the sim-K1 values reduces substantially from 4.22 to 1.78 (for discarded images). Additionally, we compare the performance in providing automated diagnoses, achieving a 7.69% improvement in accuracy, from 81.32% using SmartKC [9] (Sens. = 87.27%, Spec. = 72.22%) to 89.01% using *SmartKC++* (Sens. = 87.27%, Spec. = 91.67%). We believe that *SmartKC++* holds potential for large-scale keratoconus screening.

2. Related Work

Several prior methods have been proposed for portable and/or low-cost screening of keratoconus [2, 3, 18, 19]. For instance, Mahmoud and Mengash [18] proposed reconstructing the 3D corneal surface from the 2D frontal and lateral eye views. Their method classified input images into one of the four stages of keratoconus (normal, mild, moderate and severe). Kobashi et al. [16] introduced a smartphone-based keratoscope that functions by capturing "selfies" of patients. The setup consisted of two 3D-printed LED rings to generate placido rings on the cornea, and the captured images were analyzed to calculate corneal parameters such as steepest and flattest keratometry and astigmatism. Askarian et al. [3] utilize a 3D-printed image acquisition tool to capture panoramic 180°views of the eye. These images are then processed using edge detection algorithms and graded using SVMs for the presence of keratoconus. Similar to SmartKC [9], many of these methods use 3D-printed attachments with smartphones. However, our system is complementary to these methods since they do not produce corneal curvature heatmaps (axial and tangential). SmartKC [9] is the only smartphone-based corneal topographer, which provides corneal curvature heatmaps. Therefore, we build on SmartKC and improve the robustness of the system, and the quality of heatmaps generated.

It is noteworthy that while deep learning methods have been extensively explored for other ophthalmic diseases such as glaucoma [8, 23, 24] and diabetic retinopathy [4, 7], deep-learning research on keratoconus is limited due to lack

of public datasets [17]. Most prior work on DL-based keratoconus screening relies on self-collected private data [1, 10].

3. Methodology

The SmartKC system [9] consists of three main components: a placido disc attachment, a data collection app, and an image processing pipeline. The image processing pipeline comprises of several steps. First, it segments the mires in the image using traditional image processing techniques. Second, mire points are localized using a radial scanning algorithm. Finally, the 3D corneal surface is reconstructed using the Arc-Step method [15], followed by Zernike polynomials based surface fitting [5, 21]. In this work, we exclusively propose modifications to the image processing pipeline to enhance its resilience to variations in data capture, thereby refraining from hardware and data collection app changes. Below, we highlight challenges faced in the current image processing pipeline and propose corresponding solutions.

3.1. U-Net based Mire Segmentation

Problem: The mire segmentation step in SmartKC utilizes a threshold-based method applied over a fingerprint detection algorithm [12]. This approach, while straightforward, often introduces artifacts or fails to segment mires accurately (see Figure 1), limiting the robustness of the system. Given that mire segmentations is the first step of the pipeline, any failures in this step cascade further into the pipeline. Enhancing the robustness of this step is thus crucial for the overall accuracy and reliability of the system.

Proposed Approach: Deep learning approaches have shown much promise for segmentation tasks over the traditional image processing approaches like the ones used in SmartKC. Motivated by this, we set out to train a U-Net [20]-based model to achieve robust and accurate mire segmentation. U-Net models have shown high accuracy and robustness for segmentation tasks, especially those involving medical images. However, ground truth segmentation annotations for mires are required for supervised training. Manually obtaining such annotations is both laborious and costly. To address this, we leverage the current thresholding and fingerprint detection based segmentation method of SmartKC as a *noisy annotator*. Failure cases (such as Figure 1 top row) were removed manually. Further, we simultaneously apply three types of augmentations: noise additions, modifications to image sharpness, and spatial augmentations (details provided in the Appendix). Although the annotations provided by the SmartKC system can be noisy and contain errors, since these errors were not systematic, the U-Net model was able to learn a representation that demonstrated robustness to variations in input images and outperformed the thresholding based mire segmentation method of SmartKC (Figure 1 bottom row).

3.2. Clustering based Mire Localization

Problem: The radial scanning algorithm [9] used in SmartKC fails to localize mires accurately when segments of mires are absent in the raw capture (Figure 1). This algorithm functions by projecting rays from the center of the segmented image along discrete angles (0° - 360°), and numbering the mires sequentially based on observation order. However, it solely relies on the relative sequence of mires along a meridian, disregarding other spatial constraints. Consequently, points on the same mire (e.g., at X° and X+1°) may receive different mire numbers if mire segments are missing (refer Figure 1). If left uncorrected, this error propagates to subsequent mires along the same angles. Although the U-Net based segmentation model enhances the robustness of mire segmentation, it does not address cases involving missing mire segments in the captured image.

Proposed Approach: To address this issue, we propose a novel clustering algorithm for mire points, inspired by connected components in a graph. Leveraging the spatial proximity of points on the same mire, we enforce local consistency in mire numbers. This helps in identifying missing mire segments and managing them appropriately in the subsequent steps. The steps of the mire localization algorithm are detailed below.

First, we use the radial scanning algorithm to identify candidate mire points. This yields a mapping between mires and pixel locations, where each (*mire_num, angle*) ordered pair maps to a pixel location. Note that this step may output erroneous mire number assignments. Next, we construct an undirected graph, with mire points as nodes. By introducing the concept of spatial proximity, we connect two nodes with an edge if they are sufficiently close in the radial and tangential directions. We empirically set the tangential tolerance to an arc-length of 4° (2° on either side of the point) and the radial tolerance to a small δ value (of 1 pixel). Next, we identify connected components within the constructed graph. Each component represents a set of mire points indexed by (*mire_num, angle*) and corresponds to a continuous mire segment.

In the subsequent step, we rectify the mire numbers identified by radial scanning, using the spatial information embedded in the connected component. With our strict criteria for adding edges between nodes, we assume that points within the same connected component belong to the same mire. Thus, the presence of two or more unique mire numbers on a connected component indicates incorrect mire numbering within the component, leading to a systematic error (mire shifts, as observed in Figure 1) in subsequent mires. To address this, we identify the most frequently ob-

served ($mode$) mire number in the connected component and calculate $\Delta = mode - mire\_num$ for all points in the connected component. Δ is positive for missing mire segments and negative for additional artifacts present in the image. Assuming that only a fraction of the points are incorrectly numbered within the connected component, we update the $mire\_num$ for all points with the $mode$. Additionally, we propagate Δ among subsequent mire points along the same angles to rectify the systematic mire shift.

Finally, we remove mire points where more than one pixel location exists for the same (*mire_num, angle*) value. This situation may arise due to the propagation of Δ in the previous step, particularly in cases where additional artifacts are present between mires. This procedure results in a set of points where mires (even broken ones) are correctly numbered, which is subsequently used for generating the corneal surface.

3.3. Robust Arc-Step Method

Problem: Both in commercial topographers and in SmartKC, the Arc-Step method [15] combined with Zernike polynomials [5, 21, 22], is employed to reconstruct the 3D corneal surface from the input mire image, camera parameters, and placido attachment specifications. The algorithm fits a cubic polynomial to the cornea's shape along individual meridians. Along a specific meridian, the algorithm proceeds radially outward and calculates the corneal point corresponding to mire n by allowing the curvature to vary *smoothly* between mire $n-1$ and mire n. However, a problem arises as the mire localization method proposed earlier allows for missing points in some mires, i.e., it is not necessary for points to be present for all angles in all mires. Hence, the corneal point for mire $n-1$ may not be available at the time of calculating mire n, rendering the Arc-Step method incompatible with the proposed mire localization approach.

Proposed Approach: To address this issue, we modify the Arc-Step method to accommodate missing points on the corneal surface. Instead of restricting the calculation to the corneal point for mire $n-1$ along the meridian, our proposed modification involves considering the corneal point corresponding to the last available inner mire for mire n (i.e., $x = max([i]); \forall\ i < n$). The Arc-Step equations are modified accordingly to take this varying distance between subsequent mires into account. Note that this method differs fundamentally from the existing approach in SmartKC [9], which extrapolates for missing information in the original input mire image during the Arc-Step method itself. In our method, the extrapolation of missing information is instead done during the Zernike polynomials fitting phase. Since the Arc-Step method is 1-D, it has very limited information from neighbors; compared to the Zernike fitting method which can take into account all 2-D neighbors for the extrapolation, and thus yield much superior fitting.

4. Experiments

4.1. Datasets

To develop and evaluate *SmartKC++*, we conducted a single-center study, collecting eye images from patients at a local eye hospital in *Anonymous City*. The eye hospital is a leading eye care and teaching institution in the area, consisting of >15 eye doctors (including >3 cornea specialists) and >10 optometrists, and treats over 500 patients every day. The study was approved by the hospital's Institutional Review Board (IRB) and informed verbal consent was obtained from all patients prior to enrollment in the study. The data collection protocol is outlined as follows:

- **Patient Selection**: Patients were evaluated by ophthalmologists using slit lamps. Those exhibiting symptoms of keratoconus were recommended for inclusion in the study, while some patients without symptoms were randomly included as controls.
- **Corneal Topography Scans**: The Optikon Keratron device, a medical-grade corneal topographer, was used as the ground truth device. It generated axial and tangential heatmaps and outputted Sim-K values, which were utilized by the ophthalmologist for keratoconus diagnosis.
- **Data Collection**: The SmartKC data collection app [9] was used to collect data (one mire image per eye) for our study.

The final dataset consists of 303 smartphone images (captured using the SmartKC app) and curvature heatmaps (axial and tangential, generated using the Optikon Keratron) from 163 patients. Among them, 133 (43.89%) were diagnosed with keratoconus and 170 (56.11%) were classified as normal, based on an assessment of the Keratron heatmaps by an ophthalmologist. The data collection was conducted in two phases, with 196 images (62 keratoconus, 134 normal) collected from 109 patients in Phase-1 and 107 images (71 keratoconus, 36 normal) from 54 patients in Phase-2.

SmartKC++ was tuned and validated with the Phase-1 data. Since images of both eyes (left and right) of the patients were collected, the dataset split for training and validation was done at the patient level to prevent the model from being trained on one eye of a patient and evaluated on the other. We report the mean results of the method on five different train/validation splits of the Phase-1 data. Additionally, to evaluate the generalization of models trained on the Phase-1 dataset, we use the Phase-2 data as a test set. Within both these sets, there are instances where the SmartKC image processing pipeline [9] fails (14 images in Phase-1 and 16 images in Phase-2), leading to artificially

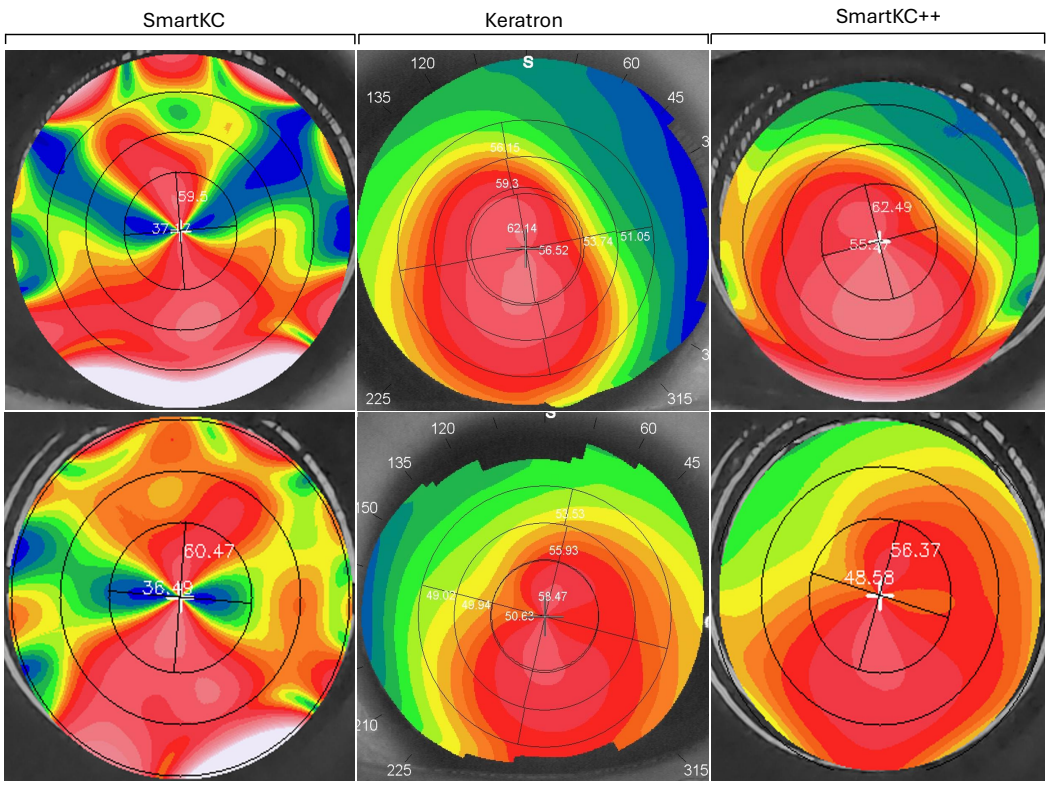

Figure 2. Qualitative improvement: Heatmaps generated by *SmartKC++* closely resemble the Keratron heatmaps compared to those output by SmartKC.

high Sim-K values and artifacts in the heatmaps. We identified such cases manually. These images, contributing to the discard rate in SmartKC, are identified as the *failure* subset of SmartKC. The remaining images, on which SmartKC generates reasonable heatmaps, constitute the *success set*. Reporting combined results on both subsets leads to unfairly high error rates for SmartKC. Hence, we report the results on the success and failure subsets separately for a fair comparison.

In summary, we report results on two datasets: Phase-1 and Phase-2. Phase-1 is a cross-validation dataset, while Phase-2 is a purely evaluation set. Both these datasets are split into *success* and *failure* subsets, based on SmartKC's performance.

4.2. Evaluation

We evaluate *SmartKC++* on three key aspects: (A) Visual similarity between heatmaps: We present examples of visual improvements in the output heatmaps resulting from our proposed changes, demonstrating higher visual similarity with the ground truth Keratron heatmaps. (B) Error in Sim-K values: The error in Sim-K values is measured using three metrics: the Pearson correlation coefficient, the mean absolute error (MAE), and the mean absolute percentage error (MAPE) between the predicted and Keratron ground truth values. (C) Accuracy of automated diagnoses: Automated diagnoses are obtained by thresholding the Sim-K values, ensuring the interpretability of the diagnoses. To identify the appropriate threshold, we employ a random forest model consisting of three decision trees. The input features for this model are Sim-K1, Sim-K2, and the difference between Sim-K1 and Sim-K2. The diagnostic performance is evaluated using six metrics: sensitivity, specificity, prediction accuracy, precision, recall, and F1-score.[2]

4.3. Results

4.3.1 Heatmap Similarity

Figure 2 showcases the visual improvements in two input images. SmartKC exhibits errors in segmenting and localizing mires, leading to artifacts in the heatmaps. *SmartKC++* addresses these issues, resulting in heatmaps closely resembling Keratron heatmaps.

[2]Code implementation of the method will be made publicly available.

		Sim-K1			Sim-K2		
		MAE	**MAPE**	**Corr.**	**MAE**	**MAPE**	**Corr.**
Success Set	**SmartKC**	3.18	**6.42**	0.81	3.17	6.63	0.58
	SmartKC++ (ours)	**3.17**	6.48	**0.83**	**2.97**	**6.32**	**0.71**
Failure Set	**SmartKC**	4.67	9.46	0.38	5.80	12.31	0.13
	SmartKC++ (ours)	**3.33**	**6.83**	**0.85**	**2.96**	**6.34**	**0.77**

Table 1. Sim-K prediction on Phase-1 dataset. Agreement between the predicted Sim-K values and those obtained from Keratron are shown. The results highlighted in bold are the best in the corresponding subset (success vs failure). *SmartKC++* outperforms SmartKC [9] in both aspects: prediction error and correlation. The means of 5 independent runs of the method are reported here. The standard deviations are reported in the supplementary material. The failure and success sets are as defined in section 4.1.

		Sim-K1			Sim-K2		
		MAE	**MAPE**	**Corr.**	**MAE**	**MAPE**	**Corr.**
Success Set	**SmartKC**	**1.29**	**2.64**	0.89	2.25	4.82	0.066
	SmartKC++	1.33	2.72	**0.92**	**1.38**	**3.04**	**0.78**
Failure Set	**SmartKC**	4.22	7.49	0.62	8.22	16.15	-0.25
	SmartKC++	**1.78**	**3.03**	**0.925**	**2.01**	**3.94**	**0.928**

Table 2. Sim-K prediction on Phase-2 dataset. Agreement between the predicted Sim-K values and those obtained from Keratron are shown. *SmartKC++* maintains a similar performance as SmartKC [9] on the success set, but significantly outperforms SmartKC on the failure set. The proposed method also significantly improved the Sim-K2 correlation across both subsets. The failure and success sets are as defined in section 4.1.

4.3.2 Errors in Sim-K

Tables 1 and 2 show the performance comparison of the proposed method on the Phase-1 and Phase-2 datasets respectively. *SmartKC++* outperforms SmartKC on both the success and failure sets. Unlike SmartKC, which struggles with the failure set, *SmartKC++* maintains robustness and decreases the Sim-K1 MAPE from 7.49% to 3.03% thereby, demonstrating consistent performance across both success and failure subsets. Notably, *SmartKC++* also significantly improves the correlation of Sim-K1 on the failure subset and that of Sim-K2 across both subsets.

4.3.3 Automated Diagnosis

Tables 3 and 4 show the performance of *SmartKC++* on the Phase-1 and Phase-2 subset, respectively, in automatically diagnosing keratoconus, by thresholding the Sim-K1 and Sim-K2 values. *SmartKC++* outperforms SmartKC by 2.92% on the Phase-1 success set, by 19.04% on the Phase-1 failure set and by 7.69% on the Phase-2 success set. As mentioned earlier, the condition for a positive keratoconus diagnosis is identified by fitting a random forest with 3 decision tree estimators, using Sim-K1, Sim-K2, and the difference between Sim-K1 and Sim-K2 as features. In the experiments, we observed that Sim-K2 was not used in the decision process, with either Sim-K1 or the difference between Sim-K1 and Sim-K2 often being repeated in one of the estimators. Hence, the condition for keratoconus diagnosis only consists of these two features. The specific threshold values were calculated as the mean of the thresholds learnt by the decision trees for five separate runs on their respective Phase-1 training sets. *SmartKC++* achieves a sensitivity of 90.00%, a specificity of 80.00%, and a prediction accuracy of 85.71% on the Phase-1 failure set, whereas SmartKC achieves a sensitivity of 85.71%, a specificity of 40.00%, and an accuracy of 66.67%. Since SmartKC fails on the failure set, the corresponding Sim-K values lack diagnostic value, thereby exhibiting poor performance. It must be noted that the failure set of the Phase-2 dataset only consisted of keratoconus positive images, hence the accuracy analysis for the failure set is skipped in Table 4. On the success set, *SmartKC++* outperforms SmartKC by 7.69% accuracy points, and exhibits higher specificity, precision, and F1 score. It is important to note that although *SmartKC++* outperforms Keratron, further large-scale evaluations are necessary to establish a reliable trend between Keratron and smartphone-based corneal topographers.

4.3.4 Ablation Study

We assess the effectiveness of the individual components in our proposed system: the U-Net based Mire Segmentation model (UbMS) and the Clustering based Mire Localization (CbML) method. The robust Arc-Step method is an accommodation for allowing broken mires, and hence is included with the mire localization method. Directly evaluating these components is challenging due to the absence of ground truth (manual mire annotation is time-taking and expensive,

Dataset	Device	KT Condition	Acc.	Sens.	Spec.	Prec.	Recall	F1
Combined	**Keratron**	K1 >49.995 or K1 - K2 >1.523	84.07	94.59	84.07	68.63	94.59	79.54
Success Set	**SmartKC**	K1 >44.55 or K1 - K2 >2.644	93.74	90.00	94.96	85.97	90.00	87.86
	SmartKC++ (ours)		**96.66**	**91.67**	**98.33**	**95.00**	**91.67**	**93.26**
Failure Set	**SmartKC**	K1 >44.55 or K1 - K2 >2.644	66.67	85.71	40.00	66.67	85.71	75.00
	SmartKC++ (ours)		**85.71**	**90.00**	**80.00**	**85.83**	**90.00**	**87.79**

Table 3. Automated diagnosis on Phase-1 dataset: Accuracy of automated diagnosis obtained by thresholding the Sim-K values. The results highlighted in bold are the best results in the corresponding subset (success / failure). *SmartKC++* outperforms SmartKC on all metrics. Notably, *SmartKC++* significantly improves the performance on the failure subset. Only the mean value of 5 runs is reported in the table. The standard deviation is provided in the supplementary material.

Dataset	Device	KT Condition	Acc.	Sens.	Spec.	Prec.	Recall	F1
Combined	**Keratron**	K1 >46.995 or (K1 - K2) >1.523	80.37	85.71	70.27	84.51	85.71	85.11
Success Set	**SmartKC**	K1 >44.35 or K1-K2 >2.644	81.32	87.27	72.22	82.76	87.27	84.96
	SmartKC++		**89.01**	**87.27**	**91.67**	**94.12**	**87.27**	**90.57**

Table 4. Automated diagnosis on Phase-2 dataset: Accuracy of automated diagnosis obtained by thresholding the Sim-K values. *SmartKC++* outperforms SmartKC on all metrics. Note that the Phase-2 failure dataset consisted of only keratoconus positive images, hence the analysis for the failure set has been skipped here.

UbMS	CbML	Acc.	F1	MAE K1	MAPE K1	MAE K2	MAPE K2
✗	✗	66.67	75.00	4.67	9.46	5.80	12.31
✓	✗	84.29	86.00	3.79	7.56	3.19	6.82
✗	✓	85.71	87.50	3.42	6.93	3.15	6.70
✓	✓	**85.71**	**87.79**	**3.34**	**6.83**	**2.96**	**6.34**

Table 5. Performance of *SmartKC++* with and without various components on the Phase-1 failure dataset. UbMS: U-Net based Mire Segmentation, CbML: Clustering based Mire Localization. The mean value of 5 runs is reported. The standard deviation is provided in the supplementary material.

hence not feasible). Instead, we indirectly evaluate their effectiveness through an ablation study, as presented in Table 5. We highlight that both components are essential to the improved performance of *SmartKC++*, with a combination of both components achieving the best performance.

5. Conclusion

In this paper, we propose *SmartKC++* to enhance the accuracy and robustness of SmartKC [9] by improving mire segmentation, localization, and the 3D corneal surface reconstruction method. We train and evaluate the proposed enhancement on a real-world dataset of 303 eye images. In an evaluation with 107 test images, we obtain an accuracy of 89.01% (sensitivity = 87.27% and specificity = 91.67%) on automated diagnosis, which is 7.69% better than SmartKC in detecting keratoconus in an automated manner. *SmartKC++* also achieves reduced errors in the reported Sim-K values. We believe our system holds great promise for mass screening of keratoconus, especially in remote and resource-constrained areas.

References

[1] Zhila Agharezaei, Reza Firouzi, Samira Hassanzadeh, Siamak Zarei-Ghanavati, Kambiz Bahaadinbeigy, Amin Golabpour, Reyhaneh Akbarzadeh, Laleh Agharezaei, Mohamad Amin Bakhshali, Mohammad Reza Sedaghat, and Saeid Eslami. Computer-aided diagnosis of keratoconus through vae-augmented images using deep learning. *Scientific Reports*, 13(1):20586, Nov 2023. 3

[2] Behnam Askarian, Jo Woon Chong, Fatemehsadat Tabei, and Amin Askarian. An affordable and easy-to-use diagnostic method for keratoconus detection using a smartphone. In Kensaku Mori and Nicholas Petrick, editors, *Medical Imaging 2018: Computer-Aided Diagnosis*. SPIE, Feb. 2018. 1, 2

[3] Behnam Askarian, Fatemehsadat Tabei, Grace Anne Tipton, and Jo Woon Chong. Novel keratoconus detection method using smartphone. In *2019 IEEE Healthcare Innovations and Point of Care Technologies, (HI-POCT)*, pages 60–62, 2019. 2

[4] Valentyn Boreiko, Indu Ilanchezian, Murat Seçkin Ayhan, Sarah Müller, Lisa M. Koch, Hanna Faber, Philipp Berens, and Matthias Hein. Visual explanations for the detection of diabetic retinopathy from retinal fundus images. In Lin-

wei Wang, Qi Dou, P. Thomas Fletcher, Stefanie Speidel, and Shuo Li, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2022*, pages 539–549, Cham, 2022. Springer Nature Switzerland. 2

[5] Luis Alberto Carvalho. Accuracy of Zernike Polynomials in Characterizing Optical Aberrations and the Corneal Surface of the Eye. *Investigative Ophthalmology & Visual Science*, 46(6):1915–1926, 06 2005. 3, 4

[6] F. Cavas-Martínez, E. De la Cruz Sánchez, J. Nieto Martínez, F. J. Fernández Cañavate, and D. G. Fernández-Pacheco. Corneal topography in keratoconus: state of the art. *Eye and Vision*, 3(1):5, Feb 2016. 1

[7] Haoxuan Che, Yuhan Cheng, Haibo Jin, and Hao Chen. Towards generalizable diabetic retinopathy grading in unseen domains. In Hayit Greenspan, Anant Madabhushi, Parvin Mousavi, Septimiu Salcudean, James Duncan, Tanveer Syeda-Mahmood, and Russell Taylor, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2023*, pages 430–440, Cham, 2023. Springer Nature Switzerland. 2

[8] Xiangyu Chen, Yanwu Xu, Shuicheng Yan, Damon Wing Kee Wong, Tien Yin Wong, and Jiang Liu. Automatic feature learning for glaucoma detection based on deep learning. In Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 669–677, Cham, 2015. Springer International Publishing. 2

[9] Siddhartha Gairola, Murtuza Bohra, Nadeem Shaheer, Navya Jayaprakash, Pallavi Joshi, Anand Balasubramaniam, Kaushik Murali, Nipun Kwatra, and Mohit Jain. Smartkc: Smartphone-based corneal topographer for keratoconus detection. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 5(4), dec 2022. 1, 2, 3, 4, 6, 7, 9

[10] Siddhartha Gairola, Pallavi Joshi, Anand Balasubramaniam, Kaushik Murali, Nipun Kwatra, and Mohit Jain. Keratoconus classifier for smartphone-based corneal topographer. *Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, 2022:1875–1878, July 2022. 3

[11] NikhilS Gokhale. Epidemiology of keratoconus. *Indian Journal of Ophthalmology*, 61(8):382, 2013. 1

[12] Lin Hong, Yifei Wan, and A. Jain. Fingerprint image enhancement: algorithm and performance evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8):777–789, 1998. 3, 9

[13] Jost B. Jonas, Vinay Nangia, Arshia Matin, Maithili Kulkarni, and Krishna Bhojwani. Prevalence and associations of keratoconus in rural maharashtra in central india: The central india eye and medical study. *American Journal of Ophthalmology*, 148(5):760–765, 2009. 1

[14] Robert H. Kennedy, William M. Bourne, and John A. Dyer. A 48-year clinical and epidemiologic study of keratoconus. *American Journal of Ophthalmology*, 101(3):267–273, 1986. 1

[15] S A Klein. A corneal topography algorithm that produces continuous curvature. *Optom. Vis. Sci.*, 69(11):829–834, Nov. 1992. 3, 4

[16] Hidenaga Kobashi, Kazuo Tsubota, Shunsuke Aoki, Masaaki Kobayashi, Brian Sumali, and Yasue Mitsukura. Evaluation of a new portable corneal topography system for self-measurement using smartphones: a pilot study. *Graefe's Archive for Clinical and Experimental Ophthalmology*, Feb 2024. 2

[17] Shawn R. Lin, John G. Ladas, Gavin G. Bahadur, Saba Al-Hashimi, and Roberto Pineda. A review of machine learning techniques for keratoconus detection and refractive surgery screening. *Seminars in Ophthalmology*, 34(4):317–326, 2019. PMID: 31304857. 3

[18] Hanan A. Hosni Mahmoud and Hanan Abdullah Mengash. Automated keratoconus detection by 3d corneal images reconstruction. *Sensors*, 21(7), 2021. 2

[19] Francisco Irochima Pinheiro, Anthony Andrey Ramalho Diniz, Evandro Pereira de Souza, Eugênio Pacelly Brandão de Araújo, and Heliana Bezerra Soares. Design and development of an ultraportable corneal topographer for smartphones as a low cost new tool for preventing blindness caused by keratoconus. *International Journal of Latest Research in Science and Technology*, 4:72–76, 2015. 1, 2

[20] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing. 3

[21] Victor Arni D. P. Sicam, Joris Coppens, Tom J. T. P. van den Berg, and Rob G. L. van der Heijde. Corneal surface reconstruction algorithm that uses zernike polynomial representation. *J. Opt. Soc. Am. A*, 21(7):1300–1306, Jul 2004. 3, 4

[22] Jason Turuwhenua. Corneal surface reconstruction algorithm using zernike polynomial representation: improvements. *J. Opt. Soc. Am. A*, 24(6):1551–1561, Jun 2007. 4

[23] Yanwu Xu, Stephen Lin, Damon Wing Kee Wong, Jiang Liu, and Dong Xu. Efficient reconstruction-based optic cup localization for glaucoma screening. In Kensaku Mori, Ichiro Sakuma, Yoshinobu Sato, Christian Barillot, and Nassir Navab, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2013*, pages 445–452, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg. 2

[24] You Zhou, Gang Yang, Yang Zhou, Dayong Ding, and Jianchun Zhao. Representation, alignment, fusion: A generic transformer-based framework for multi-modal glaucoma recognition. In Hayit Greenspan, Anant Madabhushi, Parvin Mousavi, Septimiu Salcudean, James Duncan, Tanveer Syeda-Mahmood, and Russell Taylor, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2023*, pages 704–713, Cham, 2023. Springer Nature Switzerland. 2