Fully Automated Soft Contact Lens Detection from NIR Iris Images

Balender Kumar1, Aditya Nigam2 and Phalguni Gupta3

1Department of Computer Science and Engineering, Indian Institute of Technology Kanpur (IITK), Kanpur, India
2School of Computer Science and Electrical Engineering, Indian Institute of Technology Mandi (IIT Mandi), Mandi, India
3National Institute of Technical Teacher’s & Research (NITTTR), Salt Lake, Kolkata, India

Keywords: Iris, Contact Lens, Hough Transform, Soft Contact Lens, Multi-scale Line Tracking.

Abstract: Iris is considered as one of the best biometric trait for human authentication due to its accuracy and permanence. However easy iris spoofing raise the risk of false acceptance or false rejection. Recent iris recognition research has made an attempt to quantify the performance degradation due to the use of contact lens. This study proposes a strategy to detect soft contact lens in visual pictures of the eye obtained using NIR sensor. The lens border is detected by considering small annular ring-like area near the outer iris boundary and locating candidate points while traversing along the lens perimeter. The system performance is evaluated over public databases such as IIITD-Cogent, UND 2010, IIITD-Vista along with our self created IITK database. The rigorous experimentation reveals the superior performance of the proposed system as compared with other existing techniques.

1 INTRODUCTION

In present world scenario, with bigger threats to the whole human race by several terrorist organizations around the world, ensuring human security is a huge challenge. Therefore automated human identification and verification are the basic requirements in order to provide secure and restricted access. There are several ways by which it can be realized such as token and knowledge based, but they can be very easily lost or circumvented. Several biometrics based solution has been deployed to ensure robust and accurate human identification and verification. Many physiological biometric traits such as face, palmprint (Nigam and Gupta, 2014b), knuckleprint (Badrinath et al., 2011; Nigam and Gupta, 2011), fingerprint, face, iris (Nigam and Gupta, 2012), ear (Nigam and Gupta, 2014c) are well suited hence studied extensively. But it is observed that no trait can adequately support and deliver a system with desired performance. Hence recently many multimodal systems are proposed (Nigam and Gupta, 2015), (Nigam and Gupta, 2014a), (Nigam and Gupta, 2013a) suggesting different combinations of knuckleprint, iris, palmprint images in order to achieve better accuracy.

Image quality is another key factor which is very relevant in such systems but its computation is very difficult as its a very subjective task. Not much work is reported, investigating the quality of iris (Nigam et al., 2013), knuckleprint (Nigam and Gupta, 2013b) and palmprint images.

Out of all the available biometric traits, arguably iris can be considered as one of the best biometric trait for human authentication process, as it contain highly distinguishable texture (Flom and Safir, 1987). Also iris pattern remain unchanged after the age of two and does not degrade over time and environment. Performance wise it is best but it is vulnerable to spoofing via printed contact lenses. Also the system performance degrades severely while subjects wear contact lens (Lovish et al., 2015; Yadav et al., 2014; Kohli et al., 2013). Contact lenses are of two types cosmetic contact lens and Non-cosmetic or soft contact lens. Soft contact lens detection is an important and challenging problem to preventing spoofing as compared to cosmetic lens due to absence of any extra texture. Very limited amount of work is done in this area. In this work we deal with detection of soft contact lens based on faint edge detection using line tracking.

There are some techniques available to detect cosmetic contact lens which is easier to discriminate. Soft contact lens are texture-less and transparent hence are very difficult to differentiate. Most of the time they are unrecognizable even by humans in NIR images. The sole available clues are faintly visible lens boundaries. Thermal images has been used...
in (Kywe et al., 2006) to detect contact lens using temperature variations. Basic edge extraction algorithms has been utilized in (Erdogan and Ross, 2013) to detect abrupt intensity changes and to extract contact lens border. Texture features has been utilized in (Yadav et al., 2014; Kohli et al., 2013) to obtain impressive contact lens detection performance. Some recent previous results are tabulated in Table 1.

<table>
<thead>
<tr>
<th>Cite</th>
<th>Technique Used</th>
<th>Database</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Kywe et al., 2006)</td>
<td>Thermo-Vision</td>
<td>39 Subject</td>
<td>50-66%</td>
</tr>
<tr>
<td>(Erdogan and Ross, 2013)</td>
<td>Edge Detection</td>
<td>ICE 2005</td>
<td>72-76%</td>
</tr>
<tr>
<td>(Yadav et al., 2014)</td>
<td>Texture Features</td>
<td>IIITD Cosent</td>
<td>67.52%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>IIITD Vista</td>
<td>65.41%</td>
</tr>
</tbody>
</table>

2 OCCLUSION EXCLUSION FROM LENS

Eyelids and eyelashes are two major challenges in detecting contact lens as they occlude significant iris regions. Particularly eyelid occlusion detection helps to set dynamic angle range which depends on the visibility of contact lens as lens cannot be beyond the eyelid boundaries. We detect occlusion from contact lens area using Canny edge detector with a high threshold (an overview of occlusion exclusion is shown in Fig. 1). Since sclera is white and texture-less there cannot be any edge points. If some edge points are present on sclera they must be due to eyelashes and eyelids. Based on these edge points dynamic angle range is defined ensuring that our algorithm never crosses the eyelid boundaries.

Iris image $I$ is segmented by algorithm proposed in (Bendale et al., 2012), that uses Hough transform and Integro differential operator to obtain center $(C_c, C_r)$ and the distance between center to limbus boundary $r$. An edge map of iris using Canny is generated using higher threshold and false edges are removed by excluding those connected component that are less than $P$ pixels in size as shown in Figure 3. Now an arrangement of scan-lines (Fig. 4) is characterized with in the radius range $[R_{\text{max}}^{L_{\text{min}}}, R_{\text{max}}^{L_{\text{max}}}])$ and $[R_{\text{min}}^{R_{\text{max}}}, R_{\text{max}}^{R_{\text{max}}}])$ for left and right iris portion respectively, as shown in Figure 5 and defined below:

$$\begin{align*}
R_{\text{max}}^{L_{\text{min}}} &= (r + \delta_1 + \text{offset}) \\
R_{\text{max}}^{L_{\text{max}}} &= (R_{\text{min}}^{L_{\text{max}}} + \delta_2) \\
R_{\text{max}}^{R_{\text{min}}} &= (r + \delta_2 + \text{offset}) \\
R_{\text{max}}^{R_{\text{max}}} &= (R_{\text{min}}^{R_{\text{max}}} + \delta_1)
\end{align*}$$

where, $\delta_1, \delta_2$ are the difference between the maximum possible radius of the lens and minimum possible radius of iris for left and right iris portions respectively. The values of $\delta_1, \delta_2$ are discussed in next subsection. The details of occlusion exclusion procedure has been given in Algorithm 1, used to estimate $\text{Top}_{\text{occ}}$ and $\text{Bottom}_{\text{occ}}$ as shown in Fig. 6.
2.1 Parameter Selection

Human iris diameter lies in the range of 10.2 to 13.0 mm with its expected value as 12 mm (Caroline and Andre, 2002) whereas soft contact lens diameter ranges from 13 mm to 15 mm (Details-a, 2010). However, diameter of an iris fluctuates from 4 to 8 mm in the dark and 2 to 4 mm in bright light (Details-b, 2010). For calculating δ₁ and δ₂, 1 mm is assumed to be equivalent to 3.779 pixels. Ideally based on heuristics δ₁ (in pixels) should be equal to 3.779 times the difference between maximum possible contact lens radius and minimum possible iris radius. But due to dark and bright/light effect it can be much more. For this work it is observed that they can vary from 25 to 35 pixel for UND, IIITD-Cogent, IIIT-Vista and IITK contact lens databases. Ideally δ₁ = δ₂, but the lens may be misplaced or shifted towards left or right. In case of right shifting, δ₂ is greater than δ₁ and vice-versa. Also, empirically δ₃ and δ₄ values depending

Algorithm 1: Occlusion Exclusion from Lens Area.

Require: I: Iris image, (Cᵢ, Cⱼ, rᵢ): center and radius.
Ensure: Bottom_occ: Minimum range of θ, Top_occ: Maximum range of θ.

Part A: Define
1: \( \theta_{\text{min}}^L, \theta_{\text{max}}^L : (5 + \frac{\pi}{4}, \frac{\pi}{4}) \) \( \text{//min angle for left,right} \)
2: \( \theta_{\text{min}}^R, \theta_{\text{max}}^R : (7 + \frac{\pi}{4}, \frac{3\pi}{4}) \) \( \text{//max angle for left,right} \)
3: \( R_{\text{min}}^L : (r_i + \delta_1 + \text{offset}) \) \( \text{//min radius for left} \)
4: \( R_{\text{max}}^L : (r_i + \delta_1) \) \( \text{//max radius for left} \)
5: \( R_{\text{min}}^R : (r_i + \delta_2 + \text{offset}) \) \( \text{//min radius for right} \)
6: \( R_{\text{max}}^R : (r_i + \delta_2) \) \( \text{//max radius for right} \)
7: S: S is 1-D array used to store current scan line for corresponding \( \theta \), for given range.
8: \( H(\theta) \): H is 1-D Array to store number of non zero entries corresponding scan line.

Part B: Steps
9: \( I_E \): Edge map of (I) using canny edge detector
// False edges are eliminated from an edge map by removing small connected component (having pixels less than \( P \)).
10: for \( \theta = \theta_{\text{min}}^L: \theta_{\text{max}}^L \) do
11: \( \text{// } \theta_{\text{min}}^L = \theta_{\text{min}}^R, \theta_{\text{max}}^L = \theta_{\text{max}}^R \text{ for left side and } \theta_{\text{min}}^R = \theta_{\text{min}}^L, \theta_{\text{max}}^R = \theta_{\text{max}}^L \) for right side.
12: \( \text{count} = 0; \)
13: for \( r = R_{\text{min}}^L: R_{\text{max}}^L \) do
14: \( \text{// } R_{\text{min}}^L = R_{\text{min}}^R, R_{\text{max}}^L = R_{\text{max}}^R \text{ for left side and } R_{\text{min}}^R = R_{\text{min}}^L, R_{\text{max}}^R = R_{\text{max}}^L \text{ in case of right side.} \)
15: \( a = C_i + r \times \cos \theta \)
16: \( b = C_i + r \times \sin \theta \)
17: \( \text{if}(I_E(a,b)) = 0) \)
18: \( \text{count} = \text{count} + 1 \)
19: end if
20: end for
21: \( \text{if}(\text{count} \geq T) \) \( \text{// } T \text{ is threshold which varies database to database.} \)
22: \( H(\theta) = 1 \)
23: else
24: \( H(\theta) = 0 \)
25: end for
26: \([\text{Bottom_occ,Top_occ}] = \text{MaxMargin}(H) \)
27: \( \text{// Find the index in array } H \text{ containing the maximum number of consecutive zeros and return its corresponding index angles} \)
28: return \( (\text{Bottom_occ,Top_occ}) \)

on visible part of contact lens can be fixed. Experimentally they are found to follow: \( \text{off set} = 0 \) and \( \delta_i, \delta_4 \in \{1,10\} \). If \( \delta_1, \delta_3 \) are greater than 10, then one have to shrink our angle range because we are going toward medial canthus or lateral canthus as shown in Figs 2, 5 which will degrade the accuracy substantially.
These scan-lines are within a radius range as discussed above, as well as they are also with an angular region ranging from \( \theta_{min}^L = \frac{5\pi}{4} \) to \( \theta_{max}^L = \frac{7\pi}{4} \) and \( \theta_{min}^R = \frac{\pi}{4} \) to \( \theta_{max}^R = \frac{3\pi}{4} \) for left and right iris respectively, with an angular distance of 1° between any two consecutive scan lines (as shown in Figure 5). Hence we are working on specified annular region of the binary edge-map to estimate the top and bottom occlusion.

**Observation:** The scan-line over sclera have very few edge pixels, on the other hand if current scan-line intersects eyelashes or eyelid then it is bound to have non-zero edge pixels. Hence for each scan-line (say at an angle \( \theta \)) count the number of edge (non-zero) pixels. If this count is greater then a threshold \( T \) then that scan-line is occluded else it is not occluded.

In order to estimate top and bottom occlusion angle, we computed two maximally distant (in terms of angular distance) scan-lines between which each and every scan-line is having edge pixels less than \( T \) (i.e. not occluded). The lower angle is called \( \text{Bottom}_\text{occ} \) which represent lower eyelid or eyelashes and the higher index angle called \( \text{Top}_\text{occ} \) which represent the upper eyelid or eyelashes as shown in Figure 6.

### 3 SOFT CONTACT LENS DETECTION

Soft contact lens detection is very challenging and only available hint is the faint lens boundaries. Hence the proposed SCLD algorithm uses Multi Scale Line Tracking (MSLT) Algorithm (Vlachos and Dermatas, 2010), which was initially used to segment retinal vessels. This algorithm can detect very faint edges and can extract soft contact lens boundaries. The steps involved in MSLT algorithm are given in Algorithm 2.

Output of MSLT gives lines of variable size and diameter based on the visibility of the contact lens border. MSLT algorithm returns a binary image in which the lens border lines are clearly visible on the sclera portion, if contact lens is present as shown in Figures 7 and 8. There must be some edge-lines due to the presence of eyelid/eyelashes. Hence occlusion exclusion from contact lens area is done as discussed in Section 2. After applying MSLT algorithm we compute two feature viz. Maximum edge line Maximum Hough Votes, that are used to detect contact lens as discussed below.

An arrangement of arc-lines (scan-lines are radially outward in horizontal direction Figure 4) as shown in Figure 9(a) are characterized (arc-lines are

<table>
<thead>
<tr>
<th>Algorithm 2: Steps involved in MSLT Algorithm (Vlachos and Dermatas, 2010).</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Brightness Normalization.</td>
</tr>
<tr>
<td>2. Automated selection of initial seed pixels.</td>
</tr>
<tr>
<td>3. Initialize confidence array for boundary tracking.</td>
</tr>
<tr>
<td>4. Populating confidence array by adding most suitable boundary pixels.</td>
</tr>
<tr>
<td>5. Repeat this process for each scale. (Multi-scale boundary tracking)</td>
</tr>
<tr>
<td>7. Smoothing using Median filter to remove irregularities.</td>
</tr>
<tr>
<td>8. Finally morphological directional filtering is performed in five different directions.</td>
</tr>
</tbody>
</table>

---

**Figure 7:** MSLTA algorithm based Masks (No Lens).

**Figure 8:** MSLTA based Masks (Soft Contact Lens).

---

### 3.1 Feature Computation

Out of all arc-line the one which has got the maximum
Algorithm 3: Soft Contact Lens Detection (SCLD) using Line Tracking.

Require: I: Iris image, (C_x, C_y): Iris Center, r_i: Iris radius, Bottom_occ, Top_occ.

Ensure: Two parameter, H1: Max Hough Voting

Max_linesize: Size of the contact lens in contact lens iris image.

Define:
1: \( \theta_{\text{min}}, \theta_{\text{max}}, (5 \times \frac{\pi}{4}), (\frac{\pi}{4}) \) // min angle for left and right side.
2: \( \theta_{\text{min}}, \theta_{\text{max}}, (7 \times \frac{\pi}{4}), (3 \times \frac{\pi}{4}) \) // max angle for left and right side.
3: \( R_{\text{min}}, (r_1) + \text{offset} \) // min radius for left side.
4: \( R_{\text{max}}, (r_i + \delta_1) \) // max radius for left side.
5: \( R_{\text{min}}, (r_1) + \text{offset} \) // min radius for right side.
6: \( R_{\text{max}}, (r_i + \delta_2) \) // max radius for right side.

Steps:
7: I: Apply Gaussian filter on (I)
8: H1: Apply MLSTA Algorithm(I) (Vlachos and Dermatas, 2010).
9: \( H1 = 0, \text{Max}_{\text{linesize}} = 0 \).
10: for \( r = R_{\text{min}}, R_{\text{max}} \) do // \( R_{\text{min}}, R_{\text{max}} \) in case of left side and \( R_{\text{min}}, R_{\text{max}} \) in case of right side.
11: \( \text{count} = 0 \).
12: for \( \theta = \theta_{\text{min}}, \theta_{\text{max}} \) do // \( \theta_{\text{min}}, \theta_{\text{max}} \) in case of left side and \( \theta_{\text{min}}, \theta_{\text{max}} \) in case of right side.
13: if \( \theta \geq \theta_{\text{bottom_occ}} \) and \( \theta_{\text{top_occ}} \geq \theta \) // Considering lines within the range
14: \( a = C_x + r \times \cos \theta \).
15: \( b = C_y + r \times \sin \theta \).
16: if \( (I(a, b) = 0) \)
17: \( \text{count} = \text{count} + 1 \) // Store corresponding Co-ordinate into \( x_i, y_i \)
18: \( \text{end if} \).
19: \( \text{end if} \).
20: end for
21: \( \text{Max}_{\text{linesize}} = \text{max}(\text{Max}_{\text{linesize}}, \text{count}) \) // Store corresponding \( \text{Max}_{\text{linesize}} \) Co-ordinate and \( r \) into \( X_i, Y_i, \text{Co-ordinate} \) and \( R_1 \).
22: end for
23: \( [H1] = \text{Hough Voting}(X_i, Y_i, R_1) \);
24: return \( (H1, \text{Max}_{\text{linesize}}) \).

number of edge pixels (MEP) (pixels that are probable candidates of lens) has been selected, and the value of MEP is used as our first feature. Also to ensure circular shape Hough voting is used. The center and radius for which maximum number of Hough votes (MHV) are obtained has been selected, and the value of MHV is used as the second feature. These features can be seen like a likelihood of soft contact lens. Both MEP and MHV values must be high in case of soft contact lens and vice-versa as shown in Figure 10. The overall flow diagram of the complete Soft Contact Lens Detection (SCLD) is shown in Figure 11.

Algorithm 4 can be used to perform Hough voting in order to extract features. All parameters used in this experimentation are reported in Table 2.

4 EXPERIMENTAL ANALYSIS

In this section experimental analysis of the proposed system is presented.

4.1 Dataset

The system performance is tested over IIITD (Yadav et al., 2014; Kohli et al., 2013), UND and self created contact lens databases, acquired using FA2 and
Figure 10: Pixels selected using MEP and MHV features.

Figure 11: Soft Contact Lens Detection (SCLD) using Line Tracking.

Figure 11: Soft Contact Lens Detection (SCLD) using Line Tracking.

\[\text{Algorithm 4: Hough Voting.}\]

**Require:** \((X_i, Y_i)\): Co-ordinate of the selected line, \(R_1\): Radius of selected line  

**Ensure:** \(H1\): Max Hough Voting pixel from accumulator array.

**Steps:**
1. \textbf{for} every edge pixel \((X_i, Y_i)\) \textbf{do}
2. \hspace{1em} \textbf{for} each possible radius value \(R_1\) \textbf{do}
3. \hspace{2em} \textbf{for} each possible gradient direction \(\Theta\) \textbf{do}  
   \hspace{3em} \text{or use estimated gradient at } (X_i, Y_i).
4. \hspace{3em} \(a = X_i - R \times \cos \Theta\)
5. \hspace{3em} \(b = Y_i + R \times \sin \Theta\)
6. \hspace{3em} \(H[a, b, r]++\)
7. \hspace{2em} \textbf{end for}
8. \hspace{1em} \textbf{end for}
9. \hspace{1em} \(H1 = \max(H)\)  
   \hspace{1em} \text{//Select maximum voting pixel}
10. \hspace{1em} \textbf{end for}
11. \hspace{1em} \textbf{return} \((H1)\)

**Table 2: Description of Parameter Values.**

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Parameter</th>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>\text{1. Occlusion Exclusion in Contact Lens Area.}</td>
<td>(r)</td>
<td>Iris radius (in pixel)</td>
<td>((\text{Bendale et al., 2012}))</td>
</tr>
<tr>
<td>\text{2. Soft Contact Lens Detection (SCLD) using Line Tracking.}</td>
<td>((X_i, Y_i))</td>
<td>Iris Center (in pixel)</td>
<td>((\text{Bendale et al., 2012}))</td>
</tr>
<tr>
<td></td>
<td>(\delta_1, \delta_2)</td>
<td>(\delta_1) and (\delta_2) (in pixels) are the difference between the maximum possible radius of the lens and minimum possible radius of the iris, where radius of contact lens and iris in millimeter (mm).</td>
<td>25 - 35</td>
</tr>
<tr>
<td></td>
<td>(\delta_3, \delta_4)</td>
<td>Constant fix by experimental analysis (in pixels)</td>
<td>0 - 10</td>
</tr>
<tr>
<td></td>
<td>(\text{offset})</td>
<td>Constant fix by experimental analysis (in pixels)</td>
<td>0 - 10</td>
</tr>
<tr>
<td>(R_{\text{min}}, R_{\text{max}})</td>
<td>Scan line interval</td>
<td>[\frac{\pi}{4}, \frac{3\pi}{4}] for left portion of iris [\frac{3\pi}{4}, \frac{7\pi}{4}] for right portion of iris</td>
<td></td>
</tr>
<tr>
<td>(R_{\text{min}}, R_{\text{max}})</td>
<td>Scan line size</td>
<td>[\sin(\theta) \cdot (X_i, Y_i) + \cos(\theta) \cdot (X_i, Y_i)] for left portion of iris [\cos(\theta) \cdot (X_i, Y_i) + \sin(\theta) \cdot (X_i, Y_i)] for right portion of iris</td>
<td></td>
</tr>
</tbody>
</table>

- Soft contact lens iris images \[\text{["Y"]}\]
- Colored/Textured lens iris images \[\text{["C"]}\]
- Normal iris images without lens \[\text{["N"]}\]

We have only considered no contact lens and soft contact lens images i.e. \[\text{["N","Y"]}\] classes and binary classification has been done. The features values as define earlier as MEP and MHV are computed for
every image. Their values are normalized so as to define a weighing scheme that can give a single score, ranging between \{0 to 1\}. Finally this normalized weighted score is used for contact lens detection.

### 4.2 Threshold Selection

In order to estimate the best possible thresholding parameters all databases are partitioned into two parts, training and testing. Best suited threshold value for the above mentioned normalized weighted score is computed over the training images by checking every value with in the range of \{0 to 1\}. Finally, the value of threshold at which the system performance got maximized (T_{best}), over training data, has been used for testing the proposed system over test dataset.

**Prediction:** Left over 34% data has been used as testing dataset. In the similar way as defined above, weighted normalized feature score has been calculated and compared against pre-computed threshold value T_{best}, for contact lens decision making.

The system performance is analyzed using standard parameters viz. CCR, Accuracy, FRR, FAR, EER and shown in Table 3. Accuracy is defined as \(100 - \frac{FAR + FRR}{2}\). To best of our knowledge this is the first work performing two class binary classification to detect soft contact lens over IIITD Vista, IIITD Cogent and UND database hence there is no available algorithm to compare. The EER and accuracy on these database using soft contact lens detection using Line Tracking algorithm is shown in Table 3 and in Figures 12. It is observed that proposed system can handle small amount of uneven illumination also. The EER over our database has been found to be low as it is collected under controlled environmental conditions.

### 5 CONCLUSION

In this work fully automatic soft contact lenses detection algorithm is proposed using NIR sensor. The lens border is detected by considering small annular ring-like area near the outer iris boundary and locating candidate points while traversing along the lens

<table>
<thead>
<tr>
<th>Describer</th>
<th>SCLD using Line Tracking</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database</td>
<td>CCR</td>
</tr>
<tr>
<td>IITK Left Eye</td>
<td>92.99</td>
</tr>
<tr>
<td>IITK Right Eye</td>
<td>93.39</td>
</tr>
<tr>
<td>IIITD Vista Left Eye</td>
<td>88.44</td>
</tr>
<tr>
<td>IIITD Vista Right Eye</td>
<td>90.99</td>
</tr>
<tr>
<td>IIITD Cogent Left Eye</td>
<td>85.90</td>
</tr>
<tr>
<td>IIITD Cogent Right Eye</td>
<td>89.48</td>
</tr>
<tr>
<td>UND</td>
<td>90.54</td>
</tr>
</tbody>
</table>

Figure 12: Comparative Analysis for both eyes.
perimeter. Multi-Scale Line Tracking (MSLT) based faint edge detection algorithm is used and features like number of edge pixels (MEP) and number of Hough votes (MHV) are used for classification. Experiments are conducted on publicly available IIITD-Vista, IIITD-Cogent, UND 2010 and our indigenous database. Results of the experiment indicate that proposed method outperforms previous soft lens detection techniques.

ACKNOWLEDGEMENTS

Authors would like to thank Indian Institute of Technology Mandi and IITK for providing funds, intellectual help and guidance.

REFERENCES


