# A Study of Convolution Neural Network Based Cataract Detection with Image Segmentation

Nina Sevani
Informatics Department
Krida Wacana Christian University
Jakarta, Indonesia
nina.sevani@ukrida.ac.id

Hendrik Tampubolon
Information System Department
Krida Wacana Christian University
Jakarta, Indonesia
hendrik.tampubolon@ukrida.ac.id

Jeremy Wijaya
Informatics Department
Krida Wacana Christian University
Jakarta, Indonesia
jeremy.2017tin005@civitas.ukrida.ac.id

Lukas Cuvianto
Informatics Department
Krida Wacana Christian University
Jakarta, Indonesia
lukas.2018tin008@ civitas.ukrida.ac.id

Albert Salomo
Informatics Department
Krida Wacana Christian University
Jakarta, Indonesia
albert.412019040@civitas.ukrida.ac.id

2022 IEEE International Conference on Communication, Networks and Satellite (COMNETSAT) | 978-1-6654-6030-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/COMNETSAT56033.2022.9994549

***Abstract*—Timely and precise cataract detection is crucial to managing the risk and preventing blindness for cataract's patients. This paper proposed a framework for automatic cataract detection consisting of the K-Means clustering-based segmentation (KMSeg) and Convolutional Neural Network (CNN). At first, data pre-processing was performed. Then, KMSeg is responsible for characterizing the input images into a subgroup of color. Lastly, three CNN were employed based on DCNN, ResNet18, and ResNet50 backbones for feature learning and classification task. An extensive study was examined on Fundus and Front Eye datasets with numerous experimental settings. The result shows that the proposed KMSeg-CNN is able to maintain accuracy yet provides a faster training and testing execution time across the dataset.**

***Keywords—image segmentation, K-means, CNN, cataract detection***

## I. INTRODUCTION

A cataract is one type of eye disease indicated by clouded lens manifestation [1]. This disease could be severe if not treated properly, which precedes blindness [2, 3]. There will be 40 million people across the globe are projected to suffer from losing their vision due to cataracts [4] in 2025. More specifically, 77.7 % of blindness in Indonesia was also found by the attack of cataract illness, as reported in [5]. In fact, this vision-impaired can be prevented if appropriate treatment is practiced at a very early stage of cataracts. Reliable cataract detection is indeed an essential issue to be provided by an ophthalmologist

Currently, the ophthalmologist examines the patients manually to inspect whether a person has suffered a cataract or not, which is time-consuming. Also, the lack of medical facilities, limited ophthalmologists, and not being evenly geographically distributed will lead to slowing down the medication treatment [6], particularly in remote areas. Obviously, the development of an automatic cataract screening system could help ophthalmologists speed up the screening prior to further medical treatment. In addition, a subjective observation might occur when an ophthalmologist analyzes the patient's eye photograph data because of the eye variation from the diverse people, and also, it is only based on the eye lens's opacity [7]. Therefore, developing an accurate and timely of automatically cataract detection remains challenging.

Meanwhile, emerging machine learning (ML) and advanced mobile communications technologies like 5G enable us to perform automatic image recognition with an enormous image size from remote sensors through mobile networks in many applications [8], such as remote surgery tool detection [9, 10], and ML-based retina detection [6]. Although the stream of these large images from remote devices can be done in real-time due to the capability of the 5G network, reducing the ML training and inferences time is still essential to meet timely automatic image recognition requirements [11].

In the literature, many studies have been developed to detect cataract disease automatically, from traditional ML-based to deep learning (DL) based approaches [11]. For example, the work in [12] proposed feature extraction based on the gray-level co-occurrence matrix (GLCM) then, followed by k-Nearest Neighbors in (k-NN) as the classifier. However, their model yields unsatisfactory accuracy performance. Similarly, GLCM feature extraction was also employed in [13], but a high-level feature extraction based on pre-trained ResNet was fused with GLCM to enhance Support vector Machine (SVM) accuracy performance at the top layer. The work in [12, 13] mentioned above relies on pre-defined hand-crafted features, which are inefficient and might have redundant and incomplete features. In addition to GLCM, image segmentation for blood vessel feature extraction was introduced prior to SVM classifier in [14]. Their work shows that the segmentation procedure can enhance accuracy and speed up performance in a real-time manner. However, their model is still dependent on hand-crafted features and traditional ML classifiers, which might fail to extract latent information in the features.

Thanks to the feature learning in DL allows the model to extract the feature automatically from the data. With the advantage of DL, researchers have shifted from traditional ML to more advanced DL-based models to tackle the cataract detection challenge [2, 3, 15, 16]. For instance, Deep Convolutional Neural Network (DCNN) was studied in [17] to deal with automatic cataract detection based on the hospital's real data and combined with the Fundus dataset. Also, the SqueezeNet based has developed in [6] with fewer parameters than DCNN in [17]. However, the work in [6, 17] was not addressed the time execution. Junayed et al. [16]. recently proposed CataractNet based on a tailored CNN network to deal with fewer parameters, faster running time, and accurate accuracy. Their work varied the proportion of training and the testing dataset, where 80% of training data found the best performance. All mentioned work above was still not considered the image segmentation, which might speed up the neural network time execution.

Recently, a study of image segmentation incorporated with a classifier model has been studied. In particular, for detecting fruit disease in [18], K-Means clustering was utilized to pre-characterize the original images, followed by feature extraction and an SVM classifier. Their result shows that segmentation approaches before the classifier can perform well with relatively small data. However, their work still employs traditional ML, which might limit its performance. Another work in [19] develops the model with hybrid optimal K-means clustering-based segmentation and Convolutional Neural Network (OKM-CNN) to recognize the vehicle's plate number. The accuracy is up to 0.98%. This work shows that K-

Means incorporated with a CNN-based model can enhance accuracy. Nevertheless, there were few studies examine a hybrid of K-Means with CNN on the Cataract detection problem. Motivated with the background introduced above, in this study, we aim to examine the effectiveness of the image segmentation approach prior to a CNN-Based classifier for cataract detection. We utilized K-Means based clustering (KMSeg) to deal with image segmentation based the color the cataract images prior to classification.Then, a feature learning was done with CNN-Based model. The contribution of this study are summarized as follows:

1) We showcase the effectiveness of the proposed KMSeg-CNN-Based to deal with timely and precise cataract detection. The proposed framework can speed up the time execution in terms of training and validation time.

2) Three CNN models based on the DCNN, ResNet18, and ResNet50 backbone were implemented with KMSeg. A comparative study was provided as a pilot study for decision-makers when deploying real-time automatic cataract detection

3) The proposed framework has been validated with Fundus and Front-Eye public datasets. The proposed KMSeg-ResNet18 outperforms the baseline in terms of time, precision, and F-score on the Fundus dataset. Besides, all of the CNN-based with KMSeg models display comparable results on Front-Eye datasets yet decrease the time execution drastically

The rest of this paper is organized into five sections. Section II details the related study of cataract detection in the previous work, and Section III describes the proposed framework and the methodology. Section IV showcases the experimental results. Finally, in Section V, we draw a conclusion based on our findings.

## II. Related Works

The hybrid of several methods in solving problems has been increasingly carried out by researchers in the field of machine learning in various fields. Generally, the motivation to use a combination of methods is to get more optimal system performance results. One example of combining clustering and classification methods, using SLIC-DBSCAN with CNN, is for the smoke image dataset [20]. In their paper, it was found that the use of the SLIC-DBSCAN model plus CNN as a complementary technique resulted in an increase in the results of the existing image classification. Where there is an increase in the precision value in the fire class as much as 2.49% and non-smoke as much as 3.53%. Another example of combining methods is also carried out in the field. There is also a paper that combines the Optimal K-means (OKM) algorithm with CNN in the case of vehicle plate detection. This paper states that the combination of OKM and CNN obtains an accuracy of 98.1 %, rivaling ResNet50. This explains that the combination of two machine learning methods can be used to produce a good classification model [19]

Previous research also revealed the need to combine unsupervised learning and supervised learning algorithms [16]. In the research, it was said that no technique outperforms other techniques because each technique has high accuracy in each different image for image segmentation problems. Therefore, knowledge of datasets and combining techniques is needed so that they can effectively segment. Another research is on the application of unsupervised learning algorithms, which will try to compare the testing of 3 algorithms, namely SVM (Support Vector Machine), K-means, and Enhanced K-means [14]. From the experiments conducted, it was found that Enhanced K-means resulted in a good accuracy estimate in the range of 55 – 86% where SVM in the range of 40.6 – 66.9% and K-means in the range of 49.6 – 77.5%. These results indicate that K-means can be relied upon in performing clustering, especially when tuning and improvements are made to the existing algorithm. The use of K-means as an unsupervised learning model to segment images also gives the result that the same technique cannot give the same results for all types of images [12, 13]. This study also reveals that there are 3 methods that can be used to improve segmentation results, namely by combining several segmentation techniques, tuning machine learning algorithm parameters for segmentation and applying the CNN model and then segmenting non-machine learning.

## III. Methodology

The proposed cataract detection framework is depicted in Fig.1. As seen in Fig. 1, it contains three main steps: data pre-processing, K-means clustering-based segmentation, and CNN-Based classification. First, image transformations like resizing, conversion, normalization, and augmentation were employed in the pre-processing step. Second, K-means clustering was utilized to pre-characterized the pre-processed RGB images into segmented images. Finally, the segmented images were then fed to train the CNN-based classifier. A detailed explanation for each step will be given in the following.

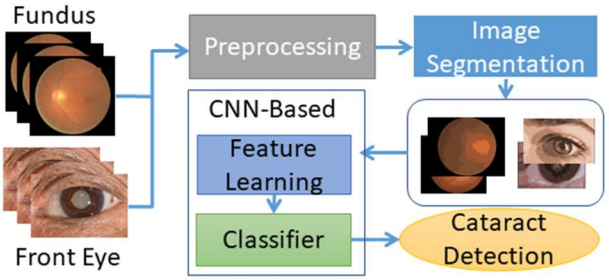


Fig. 1. The proposed cataracts detection framework overview

### A. Dataset

Two public datasets were examined to investigate the performance of the proposed framework, namely the Fundus dataset which was obtained by ophthalmoscopy [21], and the Front Eye dataset [22].. The description of the datasets is depicted in Fig.2. As shown in Fig.2., the Fundus dataset contains 100 cataract image samples as positive labels, and the rest is labeled as negative samples with a total of 400 samples. In contrast, the Front Eye Dataset contains more samples with cataract images of 3714 samples out of 8068 samples, which is considered a relatively balanced sample. Please note that the Front Eye dataset in [22] was curated from a google search that is already augmented with ten different augmentations techniques.

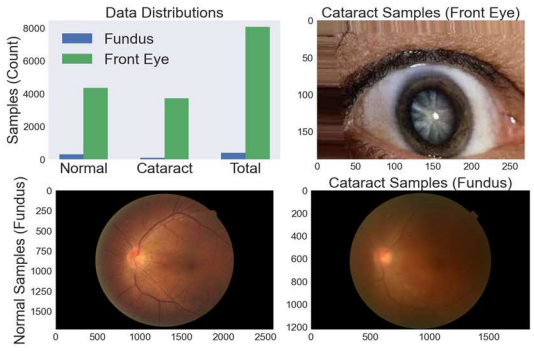


Fig. 2. Dataset proportion and examples

### B. Preprocessing

In this step, all the images were resized into 224x224 with bilinear interpolation. Also, the images were normalized into the range [0,1], subsequently standardized the images with the mean and standard deviation, which were set to [0.485, 0.456, 0.406] and [0.229, 0.224, 0.225], respectively. Image augmentation, such as

rotation, flipping, zooming, and cropping, was also done before the model training. This augmentation was employed because the number of samples in Fundus data is relatively small. Similarly, image augmentation was also used, which was already provided in the Front Eye dataset. The final image sample is transformed into a tensor shape of Nx3x224x224, where N is the number of samples

### C. Image Segmentation

The image segmentation process in the proposed framework is based on the color in the image and divided into a particular discrete color group. This step aims to reduce the noise and transform the image into a more compact image. To achieve that, we then adopted the K-Means clustering method called KMSeg. Suppose the input image has $w\ x\ h$ resolution; then image $w\ x\ h$ needs to be grouped into $k$ number of the color segment. Let the $(w_i, h_j)$ be the first point and $(c_a, c_b)$ be the second point, a candidate for the center of the color segment, $i$, and $j$ is the index. The distance of the nearest centroid can be calculated by Euclidean distance $ED$, which can be rewritten in (1).

$$ED\ [(w_i, h_j), (c_a, c_b)] = \sqrt{(w_i - c_a)^2 + (h_j - c_b)^2} \quad (1)$$

The overall step in KMSeg can be seen in Fig. 3. At the initial step, the $k$ should be determined, such as $k = 5$. For each image samples take each R, G, and B channel into a 2D array, then assign the pixel to the closest center based on $ED$. Repeat the process until meet the threshold tolerance. Finally, reshape the grouped pixel into the original image $w\ x\ h$ size. In this work, we vary the $k$ into different scenarios from k= 4 to k=10. The result of KMSeg will then be used in the CNN-Based classification model.

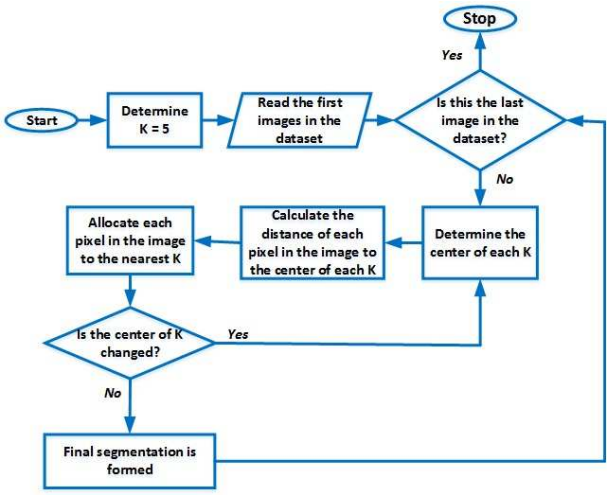


Fig. 4. KMSeg Image Segmentation Flow Diagram

### D. Convolution Neural Network Based Classification

CNN-based model literally shares the same notion, namely, feature extraction and fully connected (FC) layer. The feature extraction usually consists of convolution operation, non-linearity and pooling, considered as one block. The FC layer contains a flattened, dense layer, an activation function, and some regularization. Despite the success of CNN, determining the architectures and hyper-parameter settings can be tricky, particularly when developing CNN on different domains and data. This study examined three different CNN backbones based on DCNN [23], ResNet18 [24], and ResNet50 [24] architectures. All layers of the network architecture and parameters can be seen in the original work in [23] [24]. The difference is only in the last layer of the FC layer. The FC layer's detailed network architecture and parameters are shown in Fig. 4. Since the task was designed to recognize the cataract and non-cataract only, the binary cross

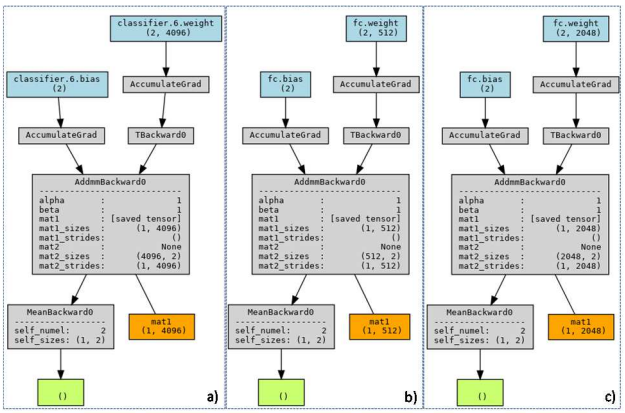


Fig. 3. Fully Connected Layer of CNN-Based Architectures in Top Layer After Feature Extraction. a) DCNN b) ResNet18 c) ResNet50

entropy (BCE) is employed. The Stochastic Gradient Descent (SGD) optimization was taken into account.

In general, the model complexity of the deep learning model can be approximated by the number of parameters (params) and floating point operations (FLOPs). A more advanced indicator of FLOPs, the multiply-accumulate operations (MACs), was also introduced in the literature. The computational cost and complexity of the three CNN-based cataract detection models used in this work can be seen in TABLE 1.

TABLE 1 CNN-BASED MODEL MEMORY COST AND COMPLEXITY

| Model | Input Resolution | No. of Params (millions) | MACs (G) |
|---|---|---|---|
| DCNN | 224x224 | 57.470 | 0.7114 |
| ResNet18 | 224x224 | 11.180 | 1.82 |
| ResNet50 | 224x224 | 23.510 | 4.12 |
| DCNN* | 224x224 | 0.00819 | 0.7114 |
| ResNet18* | 224x224 | 0.00103 | 1.82 |
| ResNet50* | 224x224 | 0.0041 | 4.12 |
| KMSeg-DCNN | 224x224 | 57.012 | 0.7101 |
| KMSeg-ResNet18 | 224x224 | 11.178 | 1.824 |
| KMSeg-ResNet50 | 224x224 | 23.512 | 4.132 |

* when pre-trained model used

### E. Experimental Settings

All the experiments were conducted under Ubuntu 20.04 environment with the following hardware: Intel Xeon E5-2630 with 2.20 GHz CPU, 32 GB RAM, and NVIDIA RTX 1080Ti of GPU. The development of image pre-processing, augmentation, and model implementation was done with python 3.7, OpenCV, and PyTorch library. In the proposed framework, all the models were trained using the SGD optimizer with a learning rate set to be 0.001, 10 batches, the number of the epoch is 30 epochs. We use the same settings on both Fundus and Front Eye datasets. 80% of the data is utilized for training and the rest for validation dan testing.

### F. Evaluation Performances

In order to evaluate the effectiveness of the proposed framework, accuracy, precision, recall, and F-score were examined as the measurement metrics. Supposedly the CNN-Based model attempt to classify whether the given image is a cataract or non-cataract. This classification task can be seen as positive or negative. In the validation phase, the actual cataract label predicted as a cataract is called a true positive (TP); meanwhile, when mispredicted is called a false positive (FP). When an actual non-cataract is predicted as non-cataract, that is called a true negative (TN), but if mispredicted, that is called a false negative (FN). The accuracy of Acc., the

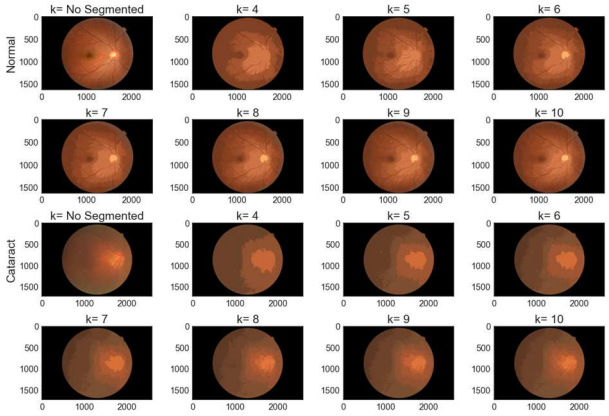


Fig. 7. Examples of segmentation result on Fundus dataset

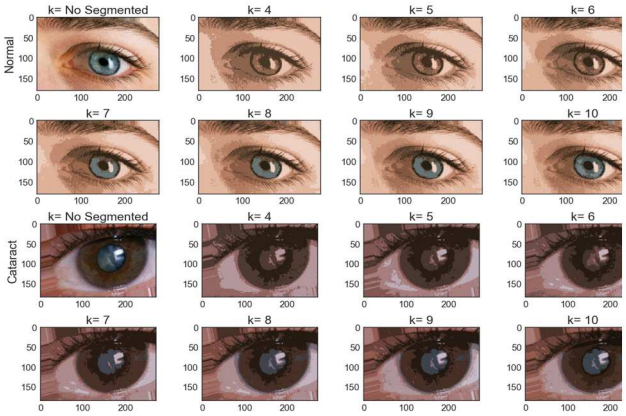


Fig. 6. Examples of segmentation result on Front Eye dataset

precision of Prec., recall of Rec., and F-score of F1 is computed as follows:

$$Acc. = \frac{(TP+TN)}{(TP+TN+FP+FN)} \quad (2)$$

$$Prec. = \frac{(TP)}{(TP+FP)} \quad (3)$$

$$Rec. = \frac{(TP)}{(TP+FN)} \quad (4)$$

$$F1 = 2\,x\frac{Prec.\ x\ Rec.}{Prec.\ +\ Rec.} \quad (5)$$

## IV. Result and Discussion

### A. Image Segmentation Result

The image segmentation has been performed on both Fundus and Front Eye datasets for all the image samples. The examples of the segmentation result with different k segment numbers are depicted in Fig. 5 and Fig. 6 for cataract and non-cataract samples, respectively. As can be seen in Fig. 5 and Fig. 6, the bigger the number of k segments, the closer the image to the original one. Also, the difference between the cataract and non-cataract samples became more evident. In the Front Eye dataset, we can notice that the color shape of the cataract eye pupil is shifted conspicuously. The fundus dataset shows that the cataract eye looks brighter in color with more lines than the non-cataract eye. This follows the morphological condition of the eye's retina, where the condition of the blood vessels and lens of the eye is different between cataract patients and non-cataract patients. Please note that segmentation prior to classification affects class membership. There might be over-segmented, which harm the small detail such as lines and artifacts, and under-segmented, which cannot restore small areas in the image. Therefore, we vary the number of segments in the training phases of CNN-Based network to investigate which number of $k$ segment will yield better accuracy.

### B. Accuracy and Time Consumption of CNN-Based Cataract Detection Model Performance

After conducting extensive experiments with a total of 30 epochs, the best model during the validation phase was then selected. The accuracy and loss of the selected models during the training and validation phases are depicted in Fig. 7. In Fig. 7, the best model of the DCNN (blue color), ResNet18 (green color), and ResNet50 (red color) with pre-trained on Fundus dataset achieve at 23, 10, and 11 of the epochs with 0.46, 0.30, 0.23 of validation loss, respectively. Meanwhile, the best model of the KMSeg-DCNN (cyan color), KMSeg-ResNet18 (magenta color), and KMSeg-ResNet50 (yellow color) were obtained when the epoch 10, 9, and 20 of epochs with loss 0.78,0.29, and 0.28. In addition, the best

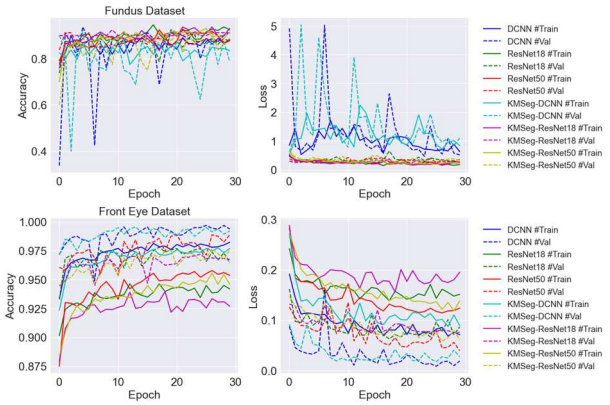


Fig. 5 Accuracy and loss graph visualization of the CNN-Based cataract detection model

model can be attained when the epochs of 25, 17, 29, 25, 28, and 26 are reached for DCNN, ResNet18, ResNet50, KMSeg-DCNN, KMSeg-ResNet18, and KMSeg-ResNet50 on Front-Eye dataset. The loss are 0.0082, 0.062, 0.039, 0.017, 0.075, and 0.058, respectively. These models were then used for more detailed analysis.

TABLE II and TABLE IV showcase the performance of the proposed KMSeg-CNN Based model on Fundus datasets. As shown in TABLE II, the execution time (training and validation process) of the CNN-Based model with KMSeg is consistently faster than the original CNN without KMSeg across the three different backbones. Based on TABLE IV, the minimum execution time needed is in KMSeg-ResNet18, which is 245.37 seconds. It means that KMSeg-ResNet18 can speed up by 0.46 times ResNet18 and 0.38 times ResNet18 with a pre-trained model. Similarly, KMSeg-DCNN show 0.39 and 0.37 times improvement for DCNN with and without the pre-trained model, respectively. Compared with the ResNet50 model, KMSeg-ResNet50 achieves 1.3 times faster than ResNet50 and 0.29 times faster than ResNet50 with pre-trained.

KMSeg-DCNN performs higher accuracy of 91.25% when the image is segmented with k of 10 clusters. However, when the pre-trained DCNN was employed, the accuracy increased up to 93.75%. KMSeg-ResNet18 shows the highest accuracy of 92.5% when the k = 5 and k=7 which also outperforms the original ResNet18. KMSeg-ResNet50 display a comparable accuracy of 91.25% with original ResNet50 when the k = 5 and k = 9. Despite the original DCNN result highest accuracy for all the cases on Fundus Dataset, however, when we examine the precision, recall, and F1 score metric, it turns out the KMSeg-ResNet18 with $k$ of 7 is superior compared to all the models, where the F1 score is achieved to be 0.928. Therefore KMSeg-ResNet18 with k = 7 is suggested in this study based on the Fundus image dataset.

TABLE II EXPERIMENTAL RESULT ON FUNDUS DATASET

| Model | w/o Seg. | | | | Model | k = 4 | | k = 5 | | k = 6 | | k = 7 | | k = 8 | | k = 9 | | k = 10 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | *t** | *Acc.** | *t* | *Acc.* | | *t* | *Acc.* | *t* | *Acc.* | *t* | *Acc.* | *t* | *Acc.* | *t* | *Acc.* | *t* | *Acc.* | *t* | *Acc.* |
| DCNN | 340.92 | **93.75** | 346.46 | 75.0 | KMSeg-DCNN | 241.43 | 88.75 | 243.49 | 88.75 | 243.97 | 88.75 | 245.44 | 88.75 | 246.59 | 88.75 | 245.83 | 90.0 | 248.08 | **91.25** |
| ResNet18 | 335.59 | 88.75 | 358.85 | 91.25 | KMSeg-ResNet18 | 240.55 | 91.25 | 243.62 | **92.5** | 243.59 | 91.25 | 245.37 | **92.5** | 244.68 | 88.75 | 246.74 | 90.0 | 258.18 | 90.0 |
| ResNet50 | 363.12 | 91.25 | 646.52 | 87.5 | KMSeg-ResNet50 | 279.96 | 90.0 | 280.51 | **91.25** | 277.73 | 88.75 | 275.58 | 90.0 | 289.72 | 90.0 | 286.12 | 91.25 | 266.91 | 90.0 |

Acc. = Accuracy in percentage unit, t = time needed for training and validation in seconds unit, * = Pre-trained model used, w/o Seg. = No image segmentation, Seg. = Image Segmentation was employed

TABLE III EXPERIMENTAL RESULT ON FRONT EYE DATASET

| Model | w/o Seg. | | | | Model | k = 4 | | k = 5 | | k = 6 | | k = 7 | | k = 8 | | k = 9 | | k = 10 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | *t** | *Acc.** | *t* | *Acc.* | | *t* | *Acc.* | *t* | *Acc.* | *t* | *Acc.* | *T* | *Acc.* | *t* | *Acc.* | *t* | *Acc.* | *t* | *Acc.* |
| DCNN | 842.35 | 99.68 | 2913.83 | 99.68 | KMSeg-DCNN | 650.08 | 98.31 | 618.84 | 99.31 | 639.34 | 99.12 | 641.67 | 99.37 | 756.32 | 99.37 | 642.01 | **99.56** | 654.02 | 99.50 |
| ResNet18 | 1716.5 | 97.81 | 5289.07 | 99.93 | KMSeg-ResNet18 | 1473.91 | 95.31 | 1372.76 | 96.81 | 1334.29 | 97.06 | 1592.25 | 96.56 | 1684.00 | **97.75** | 1315.79 | 97.56 | 1392.42 | 97.18 |
| ResNet50 | 5294.1 | 98.87 | 15161.9 | 99.87 | KMSeg-ResNet50 | 4552.36 | 96.18 | 4446.42 | 97.50 | 5007.05 | 97.87 | 4484.48 | 98.18 | 5179.43 | **98.50** | 4982.88 | 97.68 | 4746.78 | 98.37 |

Acc. = Accuracy in percentage unit, t = time needed for training and validation in seconds unit, * = Pre-trained model used, w/o Seg. = No image segmentation, Seg. = Image Segmentation was employed

TABLE IV PRECISION, RECALL, AND F1 SCORE COMPARISON ON FUNDUS DATASET

| Model | Time(s) | Acc. | Prec. | Rec. | F1 |
|---|---|---|---|---|---|
| DCNN | 346.46 | 0.7500 | 0.7478 | **0.9926** | 0.8458 |
| ResNet18 | 358.85 | 0.9125 | 0.8834 | 0.9107 | 0.8834 |
| ResNet50 | 646.52 | 0.8750 | 0.8362 | 0.8493 | 0.8174 |
| DCNN* | 340.92 | **0.9375** | 0.9062 | 0.8908 | 0.8848 |
| ResNet18* | 335.59 | 0.8875 | 0.9194 | 0.9388 | 0.9208 |
| ResNet50* | 363.12 | 0.9125 | 0.9057 | 0.9425 | 0.9159 |
| KMSeg-DCNN** | 248.08 | 0.9125 | 0.8940 | 0.8809 | 0.8707 |
| KMSeg-ResNet18*** | **245.37** | 0.9250 | **0.9247** | 0.9471 | **0.9280** |
| KMSeg-ResNet50**** | 280.51 | 0.9125 | 0.9073 | 0.9427 | 0.9158 |

* is the pre-trained model used followed by a fine-tuning in the last layer, ** is segmented by k=10, *** is segmented by k=7, and **** is segmented by k=5

TABLE V PRECISION, RECALL, AND F1 SCORE COMPARISON ON FRONT EYE DATASET

| Model | Time (s) | Acc. | Prec. | Rec. | F1 |
|---|---|---|---|---|---|
| DCNN* | 842.35 | 0.9968 | 0.977 | 0.978 | 0.975 |
| ResNet18* | 1716.46 | 0.9781 | 0.941 | 0.959 | 0.941 |
| ResNet50* | 5294.08 | 0.9887 | 0.947 | 0.965 | 0.948 |
| DCNN | 2913.83 | 0.9968 | 0.965 | 0.976 | 0.966 |
| ResNet18 | 5289.07 | **0.9993** | **0.981** | **0.988** | **0.981** |
| ResNet50 | 15161.9 | 0.9987 | 0.963 | 0.975 | 0.963 |
| KMSeg-DCNN** | **642.01** | 0.9956 | 0.974 | 0.974 | 0.971 |
| KMSeg-ResNet18*** | 1392.42 | 0.9718 | 0.933 | 0.951 | 0.931 |
| KMSeg-ResNet50** | 4982.88 | 0.9768 | 0.964 | 0.968 | 0.960 |

* is the pre-trained model used then fine-tuning the last layer, ** is segmented into 9 cluster, *** is segmented into 10 cluster

In addition to the evaluation of the Fundus dataset, the performance measurement was also conducted on the Front Eye dataset, which can be seen in TABLE III and TABLE V. As shown in TABLE III, CNN-Based model with KMSeg also exhibits better time execution rather than original CNN-Based model. Based on TABLE V, the minimum execution time needed is in KMSeg-DCNN, which is 642.01 seconds, which enhances 3.53 and 0.31 times the DCNN without and with pre-trained. In contrast, ResNet50 consumes execution time the most by15161.9 and 5294.08 seconds, without and with a pre-trained model. When KMSeg is performed, the improvement can be achieved up to 2.04 and 0.06 times without and with a pre-trained model. Also, KMSeg-ResNet18 can speed up by 2.79 times ResNet18 and 0.23 times ResNet18 with a pre-trained model.

The performance of KMSeg-DCNN reaches up to 99.56 percent of accuracy when the number of k is 9. Both KMSeg-ResNet18 and KMSeg-ResNet50 show the highest accuracy when the number of k=8. As mentioned earlier, accuracy is not always the best measure of classification performance. We then validate the performance of each model with the precision, recall, and F1 score metric. The comparison result can be seen in TABLE V. ResNet18, with no pre-trained model, was used to outperform overall for the Front Eye dataset in terms of F1 score but sacrificed significance time execution. Both KMSeg-DCNN and KMSeg-ResNet50 performed better F1 when k=9 was carried out. Meanwhile, KMSeg-ResNet18 resulting a better F1 when k =10 was used. When comparing the three CNN-Based with KMSeg and without KMSeg, the F1 score is still comparable but yet reduces the time needed drastically. For the rest of the experimental results, please refer to TABLE III and TABLE V. Based on this experimental study, we suggest the KMSeg-CNN-based when time is more critical.

## V. CONCLUSION

In this paper, we demonstrated the cataract detection framework (KMSeg-CNN Based), which consists of image segmentation and CNN-based classification. At first, standard image pre-processing and augmentation were employed to tackle the small and imbalanced dataset. Then, image segmentation is applied prior to classification. Three CNN-Based (DCNN, ResNet18, and ResNet50) models were compared with the pre-trained and non-pre-trained models. Numerous of the k-segment examined from k=4 to k=10 to investigate the effectiveness of the proposed framework. Two public cataract datasets have been studied in our work. The proposed KMSeg-CNN-Based reduces the time execution drastically across the dataset, up to three times than the baseline when no pre-trained was used and 30% of the baseline when pre-trained used on the Front Eye dataset. Besides, KMSeg-CNN-Based also reduces the time consumption compared to the baseline model by up to 130% no pre-trained and 38% with pre-trained on the Fundus dataset. Also, the proposed KMSeg-CNN-Based can reach up to 0.972 of the F1-score, and 0.928 of the F1-score on Front-Eye dataset and Fundus Dataset, respectively. We hope this paper sheds light on the experimental result mentioned above when deploying automatic cataract detection.

Nevertheless, the proposed KMSeg is based on the color in the pixel. There might be a loss of information in tiny detail, such as the lines of the blood vessel. In the future, instance-based segmentation is worth investigating to excel in the model capabilities. Moreover, a more advanced of the CNN based model with a hybrid local and global segmentation, as well as an attention module, can be studied to improve the accuracy and precision of the cataract detection model performance

## REFERENCES


[1] M. K. Hasan *et al.*, "Cataract Disease Detection by Using Transfer Learning-Based Intelligent Methods," *Computational and Mathematical Methods in Medicine,* vol. 2021, p. 7666365, 2021/12/08 2021, doi: 10.1155/2021/7666365.

[2] I. Weni, P. E. P. Utomo, B. F. Hutabarat, and M. Alfalah, "Detection of Cataract Based on Image Features Using Convolutional Neural Networks," *Indonesian Journal of Computing and Cybernetics Systems),* vol. 15, no. 1, pp. 75-86, 2021.

[3] D. Kim, T. J. Jun, D. Kim, and Y. Eom, "Tournament Based Ranking CNN for the Cataract grading," *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC),* pp. 1630-1636, 2019.

[4] X. Xu, L. Zhang, J. Li, Y. Guan, and L. Zhang, "A Hybrid Global-Local Representation CNN Model for Automatic Cataract Grading," *IEEE Journal of Biomedical and Health Informatics,* vol. 24, no. 2, pp. 556-567, 2020, doi: 10.1109/JBHI.2019.2914690.

[5] T. M. o. H. Indonesia. "Data Center and Information Technology." www.pusdatin.kemkes.go.id (accessed Aug, 31, 2022).

[6] X. Qian, E. W. Patton, J. Swaney, Q. Xing, and T. Zeng, "Machine Learning on Cataracts Classification Using SqueezeNet," in *2018 4th International Conference on Universal Village (UV)*, 21-24 Oct. 2018 2018, pp. 1-3, doi: 10.1109/UV.2018.8642133.

[7] K. Y. Son *et al.*, "Deep Learning-Based Cataract Detection and Grading from Slit-Lamp and Retro-Illumination Photographs: Model Development and Validation Study," *Ophthalmology Science,* vol. 2, no. 2, p. 100147, 2022/06/01/ 2022, doi: https://doi.org/10.1016/j.xops.2022.100147.

[8] X.-L. Huang, X. Ma, and F. Hu, "Editorial: Machine Learning and Intelligent Communications," *Mob. Networks Appl.,* vol. 23, no. 1, pp. 68-70, / 2018, doi: 10.1007/s11036-017-0962-2.

[9] H. A. Hajj, M. Lamard, K. Charrière, B. Cochener, and G. Quellec, "Surgical tool detection in cataract surgery videos through multi-image fusion inside a convolutional neural network," in *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 11-15 July 2017 2017, pp. 2002-2005, doi: 10.1109/EMBC.2017.8037244.

[10] N. R. El-Far, S. Nourian, Z. Jilin, A. Hamam, S. Xiaojun, and N. D. Georganas, "A cataract tele-surgery training application in a hapto-visual collaborative environment running over the CANARIE photonic network," in *IEEE International Workshop on Haptic Audio Visual Environments and their Applications*, 1-1 Oct. 2005 2005, p. 4 pp., doi: 10.1109/HAVE.2005.1545647.

[11] A. H. Vyas and V. Khanduja, "A Survey on Automated Eye Disease Detection using Computer Vision Based Techniques," in *2021 IEEE Pune Section International Conference (PuneCon)*, 16-19 Dec. 2021 2021, pp. 1-6, doi: 10.1109/PuneCon52575.2021.9686479.

[12] N. Nur, S. Cokrowibowo, and R. Konde, "Cataract Detection in Retinal Fundus Image Using Gray Level Co-occurrence Matrix and K-Nearest Neighbor," in *International Joint Conference on Science and Engineering 2021 (IJCSE 2021)*, 2021: Atlantis Press, pp. 268-271.

[13] Y. Xiong, Z. He, K. Niu, H. Zhang, and H. Song, "Automatic cataract classification based on multi-feature fusion and SVM," in *2018 IEEE 4th International Conference on Computer and Communications (ICCC)*, 2018: IEEE, pp. 1557-1561.

[14] M. K. Behera, S. Chakravarty, A. Gourav, and S. Dash, "Detection of Nuclear Cataract in Retinal Fundus Image using RadialBasis FunctionbasedSVM," in *2020 Sixth International Conference on Parallel, Distributed and Grid Computing (PDGC)*, 2020: IEEE, pp. 278-281.

[15] Y. Dong, Q. Zhang, Z. Qiao, and J.-J. Yang, "Classification of cataract fundus image based on deep learning," in *2017 IEEE international conference on imaging systems and techniques (IST)*, 2017: IEEE, pp. 1-5.

[16] M. S. Junayed, M. B. Islam, A. Sadeghzadeh, and S. Rahman, "CataractNet: An Automated Cataract Detection System Using Deep Learning for Fundus Images," *IEEE Access,* vol. 9, pp. 128799-128808, 2021, doi: 10.1109/ACCESS.2021.3112938.

[17] M. R. Hossain, S. Afroze, N. Siddique, and M. M. Hoque, "Automatic detection of eye cataract using deep convolution neural networks (DCNNs)," in *2020 IEEE region 10 symposium (TENSYMP)*, 2020: IEEE, pp. 1333-1338.

[18] P. K. Devi and Rathamani, "Image Segmentation K-Means Clustering Algorithm for Fruit Disease Detection Image Processing," in *2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, 5-7 Nov. 2020 2020, pp. 861-865, doi: 10.1109/ICECA49313.2020.9297462.

[19] I. V. Pustokhina *et al.*, "Automatic Vehicle License Plate Recognition Using Optimal K-Means With Convolutional Neural Network for Intelligent Transportation Systems," *IEEE Access,* vol. 8, pp. 92907-92917, 2020, doi: 10.1109/ACCESS.2020.2993008.

[20] D. Sheng, J. Deng, and J. Xiang, "Automatic smoke detection based on SLIC-DBSCAN enhanced convolutional neural network," *IEEE Access,* vol. 9, pp. 63933-63942, 2021.

[21] Y. W. Chen. "cataract dataset." Kaggle. https://www.kaggle.com/datasets/jr2ngb/cataractdataset (accessed Aug 27, 2022).

[22] K. B. Ojha. "Cataract Detection Using CNN." Github. https://github.com/krishnabojha/Cataract_Detection-using-CNN/tree/master/Dataset (accessed Aug 30, 2022).

[23] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM,* vol. 60, no. 6, pp. 84–90, 2017, doi: 10.1145/3065386.

[24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 27-30 June 2016 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90.