Pattern Recognition 107 (2020) 107510

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

Efficient sampling-based energy function evaluation for ensemble optimization using simulated annealing

János Tóth*, Henrietta Tomán, András Hajdu

Faculty of Informatics, University of Debrecen 4002 Debrecen PO Box 400, Hungary

ARTICLE INFO

Article history:
Received 8 October 2019
Revised 26 May 2020
Accepted 17 June 2020
Available online 19 June 2020

2010 MSC:
68U10

Keywords:
Diabetic retinopathy
Ensemble
Microaneurysm detection
Parameter optimization
Sampling-based evaluation
Simulated annealing

ABSTRACT

In this study, we attempted to develop a method for accelerating parameter optimization of an object detector ensemble over large image datasets by using simulated annealing. We propose a novel sampling-based evaluation method that considers the minimum portion of the dataset required in each iteration to maintain solution quality. This approach can be considered a noisy evaluation of the energy. The sample sizes required during the search process are theoretically determined by adapting the convergence results for noisy evaluation. To determine applicability, we prepared and optimized two ensembles for diabetic retinopathy pre-screening based on microaneurysm detection with convolutional neural network-based and traditional object detectors. Our experimental results indicate that the proposed sampling-based evaluation method substantially reduced the computational time required for optimizing the parameters of the ensembles while preserving solution quality.

1. Introduction

Parameter optimization problems often arise in object detection when the aim is to tune adjustable parameters to maximize performance. From a theoretical point of view, the objective of optimization can be considered a function of the parameters, and the optimal settings can be found by calculating the parameter values that make the partial derivatives equal to 0. Unfortunately, this mathematical calculus technique is suitable only when the partial derivatives can be given in closed forms; thus, it is rarely useful for real-life problems. Stochastic search algorithms are commonly used to overcome this difficulty and to handle discrete problems. For example, simulated annealing (SA) [1] has very attractive properties, and this technique has been widely applied. However, performing stochastic searches may still be time-intensive because the number of parameters and their range may lead to a large, high dimensional search space, and the evaluation of complex objective functions may be very time consuming. One possible approach for addressing the latter problem involves incomplete evaluation, i.e., calculating the objective function value with some error to reduce the computational complexity.

Ensembles are often used for object detection tasks because they usually outperform their constituent members if their behavior is sufficiently diverse [2]. In these approaches, an ensemble comprises member detector algorithms with adjustable parameters, and the objective function is derived based on some aggregation rule (e.g., majority voting). Thus, the aim is optimizing the parameters of the individual members in order to maximize the performance at the ensemble level.

In a preliminary study [3], we implemented this approach for retinal image analysis by tuning the parameters of an ensemble of microaneurysm (MA) detector algorithms (a more comprehensive description of this field is provided in Section 3). We aggregated the outputs of the individual MA detectors using majority voting and measured the difference compared with the ground truth data provided with the dataset employed. The optimal parameter setting, i.e., the setting that maximized the detection accuracy, was found by SA. A bottleneck in this approach is the computational demand during the evaluation of the objective function at each search step because it requires the application of the aggregation rule to the output of the members for a parameter setting for all images in the dataset. To overcome this difficulty in our previous study [3], we successfully tested the evaluation of the objective function over only a certain subset of the dataset prepared in every search step by randomly sampling the images; however, our approach was completely heuristic regarding the level of sampling applied during the search process. Similar heuristic principles

* Corresponding author.
E-mail addresses: toth.janos@inf.unideb.hu (J. Tóth), toman.henrietta@inf.unideb.hu (H. Tomán), hajdu.andras@inf.unideb.hu (A. Hajdu).

https://doi.org/10.1016/j.patcog.2020.107510

are applied in stochastic/mini-batch gradient descent algorithms in machine learning tasks. Namely, the objective function is evaluated based on a single sample or on a subset of the training dataset with a fixed size while learning the model parameters [4].

In this study, we theoretically established a sampling-based evaluation method for SA that preserves the convergence properties of this stochastic search technique. The main contribution of our approach is recognizing that sampling can be considered a specific type of noisy evaluation [5] of the objective function. Thus, after well-designed transformations, the convergence results for noisy evaluation are suitable for sampling-based evaluation.

Our experimental results demonstrated that the proposed method significantly reduced the time required for search while also preserving the solution quality.

The remainder of this paper is organized as follows. In Section 2, we introduce our basic concepts and notations and describe our sampling strategy and the SA-based search algorithm incorporating it. Our main result regarding the determination of the minimum sample size required during the search is also formulated in this section as Theorem 1. In Section 3, we present an application to retinal image analysis for pre-screening diabetic retinopathy (DR) based on the presence of MAs. Our experimental findings regarding the classification of retinal images according to DR are presented in Section 4. Detailed results are provided in terms of the computational time reductions obtained using the proposed method while also maintaining the solution quality. We also showed that an efficient ensemble of MA detectors can be prepared for pre-screening DR. Besides, we demonstrated that the proposed method can also be used to optimize the detector ensemble for the accurate localization of MAs. Finally, we present our conclusions in Section 5.

2. SA with sampling-based evaluation

Dealing with large optimization problems typically involves making a tradeoff between accuracy and computational time. In this study, we propose an evaluation method for SA that can maintain the solution quality while reducing the runtime for objective (referred to as energy hereafter) functions that are commonly used to evaluate the average performance of object detectors and classifiers over datasets. In particular, we propose a sampling strategy that considers only a suitable portion of the dataset in each search step to maintain the convergence of SA, which approach can be considered a noisy evaluation of the energy function. The appropriate sample sizes required during the search process are theoretically determined by adapting the convergence results for noisy evaluation in SA.

2.1. Convergence of SA in the case of noisy evaluation

SA is a local search algorithm inspired by the annealing process in metallurgy, and it was introduced by Kirkpatrick et al. [6] and independently by Černý [7] to address difficult combinatorial optimization problems. The main feature of SA is the capacity to escape from local optima by accepting non-improving moves with a probability that depends on the difference in the energy function values between the current and candidate states, and a decreasing control parameter (called temperature). The method applied to generate the sequence of temperature levels is called a cooling schedule, and its choice strongly influences the performance of SA. The simplicity and general applicability of SA have resulted in this procedure being used widely to address both discrete and continuous optimization problems (for a comprehensive discussion of the theory and application of SA, see [1]).

Originally, SA was designed based on the assumption that the energy of a state can be calculated exactly, but the evaluation of a state is often subject to noise in practical problems. As a consequence, several studies have investigated the convergence properties of SA in noisy environments. The first study of this topic by Kushner [8] involved asymptotic analysis of SA under suitable conditions based on the theory of large deviations while assuming Gaussian noise. By considering discrete search spaces and assuming that the noise is normally distributed with mean 0 and variance $(\sigma^{(k)})^2 > 0$ in the kth ($k \in \mathbb{N}$) iteration, Gelfand and Mitter proved [5] that SA using noisy evaluation also converges to the globally optimal solution in probability in the same manner as that when using exact energy values if the standard deviation $\sigma^{(k)}$ of the noise is dominated by the temperature $T^{(k)}$ in the kth iteration for each k, i.e., when

$$\sigma^{(k)} = o(T^{(k)}), \tag{1}$$

where o is a Bachmann–Landau symbol that expresses a stronger requirement on the asymptotic behavior of a function than O (for further details, see [9]). Assuming the same noise properties for a specific annealing schedule, Gutjahr and Pflug [10] proved that SA converges in probability to the globally optimal solution if the standard deviation of the noise is at least inversely proportional to the number of iterations, i.e., when

$$\sigma^{(k)} = O(k^{-\gamma}) \text{ with some } \gamma > 1. \tag{2}$$

They generalized the proof of convergence to an arbitrary noise distribution that is symmetric and more peaked around 0 than the Gaussian distribution.

2.2. Basic concepts and notations

As the classic formulation, let $\mathcal{D} = \{D_1, D_2, \ldots, D_L\}$ be a set (ensemble) of $L \in \mathbb{N}$ classifiers (voters) with $D_i : \Lambda \subseteq \mathbb{R}^m \to \mathbb{R}^M_{\geq 0}$ ($i = 1, \ldots, L$), and $\Omega = \{\omega_1, \omega_2, \ldots, \omega_M\}$ is a set of finite class labels. The classifier D_i assigns the support values $D_i(\lambda) = (d_{i,1}(\lambda), \ldots, d_{i,M}(\lambda))$ to a feature vector $\lambda \in \Lambda$, which describes the opinion of the classifier in terms of the degree to which λ should be labeled by $\omega_1, \ldots, \omega_M$, respectively. Then, in a fusion-based scenario, the final class label for λ is determined by applying some aggregation rule to the individual labels supported by the classifiers $D_1, \ldots, D_L$. The simple majority voting-based classic ensemble classifier can be derived by restricting the support of the individual classifiers with $d_{i,r}(\lambda) = \delta_{jr}$, where $r = 1, \ldots, M$ if the classifier D_i labels λ in the class ω_j. The final labeling of the ensemble is based on determining the class that receives the largest support in terms of the number of votes. In our application, the population $\Lambda = \Lambda_N$ requiring classification is a dataset of N images, while the members of the ensembles are object detector algorithms and their outputs are aggregated using majority voting. Different parameter settings can be considered for these member algorithms, so we let Π_i denote the parameter domain of the classifier $D_i (i = 1, \ldots, L)$ and $\pi \in \Pi = \Pi_1 \times \Pi_2 \times \ldots \times \Pi_L$ is a given parameter vector of the ensemble. Then, the ensemble with a specific parameter setting π will be denoted by $\mathcal{D}^{(\pi)}$.

To consider the noisy evaluation of the energy, the ensemble $\mathcal{D}^{(\pi)}$ with classification accuracy $p_{\mathcal{D}^{(\pi)}} \in [0, 1]$ is a discrete random variable $X_{\mathcal{D}^{(\pi)}}$ with mean $\mathbb{E}(X_{\mathcal{D}^{(\pi)}})$ and variance $\mathrm{Var}(X_{\mathcal{D}^{(\pi)}})$, where $\mathbb{E}(X_{\mathcal{D}^{(\pi)}}) = p_{\mathcal{D}^{(\pi)}}$. Let $x^i_{\mathcal{D}^{(\pi)}}$ denote the ith realization of $X_{\mathcal{D}^{(\pi)}}$ ($i = 1, \ldots, N$). Furthermore, let the energy function E_π used to evaluate the performance of the ensemble $\mathcal{D}^{(\pi)}$ for a given parameter setting π be the empirical mean value of $X_{\mathcal{D}^{(\pi)}}$, i.e., the mean $\mu^N_{\mathcal{D}^{(\pi)}}$ of N realizations:

$$E_\pi = \mu^N_{\mathcal{D}^{(\pi)}} = \frac{1}{N} \sum_{i=1}^{N} x^i_{\mathcal{D}^{(\pi)}}. \tag{3}$$

Calculating the energy function value can be computationally expensive when considering large populations, so we estimate it using sampling. Thus, we select a random sample $|\Lambda_n| = n$ from the finite population Λ_N, i.e., $\Lambda_n \subseteq \Lambda_N$ ($n \leq N$), and for a parameter setting π estimate the corresponding energy function value with $\widehat{E}_{\Lambda_n,\pi}$ as the sample mean $\bar{x}^{\Lambda_n}_{\mathcal{D}^{(\pi)}}$ by using the following:

$$\widehat{E}_{\Lambda_n,\pi} = \bar{x}^{\Lambda_n}_{\mathcal{D}^{(\pi)}} = \frac{1}{n} \sum_{j:\lambda_j \in \Lambda_n} x^j_{\mathcal{D}^{(\pi)}}. \tag{4}$$

If a parameter setting π is fixed, then we use the brief notations E and $\widehat{E}_{\Lambda_n}$ instead of E_π and $\widehat{E}_{\Lambda_n,\pi}$, respectively.

As a special case, in a binary classification problem, the ensemble $\mathcal{D}^{(\pi)}$ with classification accuracy $p_{\mathcal{D}^{(\pi)}}$ is a random variable $X_{\mathcal{D}^{(\pi)}}$ from a Bernoulli distribution with

$$P(X_{\mathcal{D}^{(\pi)}} = 1) = p_{\mathcal{D}^{(\pi)}}, \quad \text{and} \quad P(X_{\mathcal{D}^{(\pi)}} = 0) = 1 - p_{\mathcal{D}^{(\pi)}}, \tag{5}$$

where $X_{\mathcal{D}^{(\pi)}} = 1$ and $X_{\mathcal{D}^{(\pi)}} = 0$ denote correct and incorrect classification by $\mathcal{D}^{(\pi)}$, respectively. In this case, for the theoretical mean and variance of the variable $X_{\mathcal{D}^{(\pi)}}$ from a Bernoulli distribution, we have

$$\mathbb{E}(X_{\mathcal{D}^{(\pi)}}) = p_{\mathcal{D}^{(\pi)}}, \quad \text{and} \quad \mathrm{Var}(X_{\mathcal{D}^{(\pi)}}) = p_{\mathcal{D}^{(\pi)}}(1 - p_{\mathcal{D}^{(\pi)}}). \tag{6}$$

2.3. Sampling strategy and its algorithmic realization

Assuming that calculating each value $x^i_{\mathcal{D}^{(\pi)}}$ ($i = 1, \dots, N$) has the same computational cost, then calculating $\widehat{E}_{\Lambda_n}$ is n/N times less computationally expensive than calculating E, but using $\widehat{E}_{\Lambda_n}$ introduces noise in the evaluation. For a sample Λ_n, the noise d_n originating from the sampling, i.e., the sampling error of the mean, can be determined as follows:

$$d_{\Lambda_n} = \widehat{E}_{\Lambda_n} - E = \bar{x}^{\Lambda_n}_{\mathcal{D}^{(\pi)}} - \mu^N_{\mathcal{D}^{(\pi)}}. \tag{7}$$

Noise may cause SA to consider an inferior state as superior because of the imprecise evaluation of the energy function. Thus, when the noise is stronger, the search is more random and the solution quality that can be reached after a given number of steps is worse, so the convergence is slower.

According to (1), to ensure the convergence of SA in the presence of noise, a sampling strategy must be applied that is suitable for controlling the standard deviation of the noise $\sigma_{d_{\Lambda_n}}$ regarding the temperature T during the search by selecting an appropriate sample size in each search step. Thus, we must determine the maximum allowed value $\sigma^{(k)}_{d_n}$ of each $\sigma_{d_{\Lambda_n}}$ for the current temperature $T^{(k)}$ in order to find the minimum sample size required. We state Lemma 1 for this purpose. Naturally, the standard deviation of the noise will be smaller when the sample size n is closer to the population size N.

Lemma 1. *A sufficiently simple general form of $\sigma^{(k)}_{d_n}$ that maximizes its value at the temperature $T^{(k)}$ can be given as follows:*

$$\sigma^{(k)}_{d_n} \approx T^{(k)}(1 - \epsilon)^k \text{ with } 0 < \epsilon < 1. \tag{8}$$

Proof. Using (1), we find that

$$\lim_{k\to\infty} \frac{\sigma^{(k)}_{d_n}}{T^{(k)}} = 0 \tag{9}$$

must hold. To maintain the limit in (9), the sequence $\{\sigma^{(k)}_{d_n}\}$ has to be decreasing such that $\lim\limits_{k\to\infty} \sigma^{(k)}_{d_n} = 0$ and $\sigma^{(k)}_{d_n} < T^{(k)}$ for each $k \in \mathbb{N}$. Based on these conditions, a sufficiently simple general form of $\sigma^{(k)}_{d_n}$ that maximizes its value can be given as (8). □

Example 1. As an application of Lemma 1, by considering the exponential cooling schedule with

$$T^{(k)} = T^{(0)}\alpha^k \text{ with } 0 \leq \alpha \leq 1, \tag{10}$$

the maximum value of $\sigma^{(k)}_{d_n}$ can be approximated as

$$\sigma^{(k)}_{d_n} \approx T^{(0)}\alpha^k(1 - \epsilon)^k \text{ with } 0 \leq \alpha \leq 1, \text{ and } 0 < \epsilon < 1. \tag{11}$$

A similar derivation can be applied for other cooling schedules as well.

Now we can formulate our main theoretical contribution regarding how to determine the sample size during the search.

Theorem 1. *For an arbitrary cooling schedule, the minimum sample size $n^{(k)}$ required at the kth iteration to maintain the convergence of the method in probability can be estimated as*

$$n^{(k)} \approx \frac{N\sigma^2_{max}}{(N-1){\sigma^{(k)}_{d_n}}^2 + \sigma^2_{max}}, \tag{12}$$

where $\sigma_{\max}$ is the worst-case maximum value of the population standard deviation $\sigma^{\mathcal{D}^{(\pi)}}_N$, and $\sigma^{(k)}_{d_n}$ can be derived using Lemma 1.

Proof. The noise defined in (7) is actually the difference between the sample mean and its expected value (the population mean), so its standard deviation is equal to the standard deviation of the sampling distribution of the mean, i.e., the standard error of the mean $\sigma_{\bar{x}^{\mathcal{D}^{(\pi)}}_n}$. Therefore, the standard deviation of the noise can be calculated as follows:

$$\sigma_{d_n} = \sigma_{\bar{x}^{\mathcal{D}^{(\pi)}}_n} = \frac{\sigma^{\mathcal{D}^{(\pi)}}_N}{\sqrt{n}} \sqrt{\frac{N-n}{N-1}}, \tag{13}$$

where $\sigma^{\mathcal{D}^{(\pi)}}_N$ is the population standard deviation and $\sqrt{(N-n)/(N-1)}$ is the finite population correction factor.

In (13), the population standard deviation $\sigma^{\mathcal{D}^{(\pi)}}_N$ is unknown, but it can be estimated using its worst-case (maximum) value $\sigma_{\max}$. It should be noted that in this case, it is not possible to estimate the population standard deviation with the sample standard deviation because the required sample size is not yet known.

Using the maximal value of the population standard deviation, the minimum required sample size $n^{(k)}$ at the kth iteration can be determined as (12). □

Example 2. For example, considering the exponential cooling schedule given in (10) and $\sigma_{\max} = 0.5$, the minimum sample size in the kth iteration can be given as

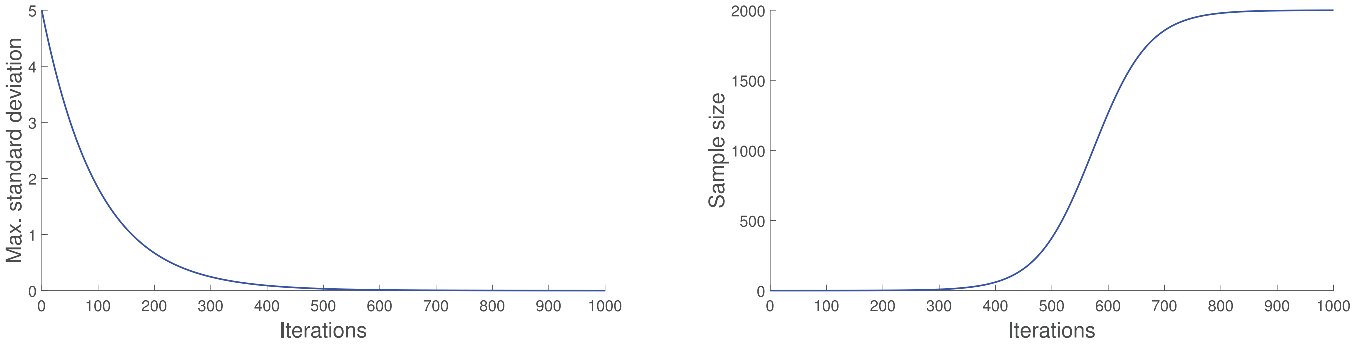
$$n^{(k)} = \frac{N}{4\,(N-1)\,(T^{(0)}\alpha^k(1-\epsilon)^k)^2 + 1}. \tag{14}$$

As a numeric demonstration for the example given above, let us consider $T^{(0)} = 5$, $k = 1000$, $\alpha = 0.99$, and $N = 2000$. For this setup, during the SA search, the maximum values allowed for the standard deviation of the noise $\sigma^{(k)}_{d_n}$ and the corresponding required sample sizes $n^{(k)}$ are shown in Fig. 1(a) and (b), respectively.

One technical issue should be noted: for every temperature value $T^{(k)}$, a minimum required sample size $n^{(k)}$ must be used; therefore, the energy function estimate of the current state should be recomputed over a sufficiently large sample in every iteration, i.e., when the temperature decreases, in order to compare the quality of the current and candidate states. However, recomputing this value would be time consuming and the evaluation would become less effective than the complete evaluation after at least half of the population is included in the sample. Therefore, in each iteration, we normalize the energy function estimate of the current state using the ratio of the minimum required sample sizes at the previous and current temperatures as

$$\widehat{E}_{norm} = \widehat{E}_{\Lambda^{(k-1)}_n} \cdot (n^{(k-1)}/n^{(k)}), \tag{15}$$

where $n^{(k-1)}$ is the sample size at which the energy function estimate $\widehat{E}_{\Lambda^{(k-1)}_n}$ of the current state is calculated and $n^{(k)}$ is the sample size at which the energy function value of the neighbor state



(a)

(b)

Fig. 1. Example of the sampling strategy in SA search with an exponential cooling schedule: (a) maximum standard deviation of the noise and (b) minimum required sample size.

will be calculated. It should be noted that the factor used for correction becomes gradually less significant as the search proceeds. As a secondary technical issue, we consider that minimum sample size should be $n \geq 50$ in order to make a reasonable assumption regarding the Gaussian distribution of the noise d_n by following the general recommendations (see [11].)

Our approach for finding the optimal parameter setting for an ensemble using the proposed sampling strategy is formally described in Algorithm 1. We refer to this algorithm as SA with Sampling-based Evaluation (SA-SBE) in the following. The algorithm contains several tunable parameters and functions, which must be selected according to the desired application. The setup corresponding to our object detection task is described in Section 3.3.

Algorithm 1 Simulated annealing with sampling-based evaluation (SA-SBE).

Input: An ensemble classifier $\mathcal{D} = \{D_1, \ldots, D_L\}$ with free parameters $\Pi = \Pi_1 \times \cdots \times \Pi_L$.
A population for classification Λ_N.
Maximum standard deviation $\sigma_{\max}$ of the energy function.
SA cooling schedule with initial temperature $T^{(0)}$.
Output: Optimal parameter setting $\pi \in \Pi$ for $\mathcal{D}$.

```
$k \leftarrow 0$
$\pi \leftarrow \text{RAND}(\Pi)$
$n \leftarrow \text{SAMPLE\_SIZE}(T^{(0)}, k, \sigma_{\max})$
$\Lambda_n \leftarrow \text{TAKE\_SAMPLE}(\Lambda_N, n)$
$\widehat{E}_{\Lambda_n,\pi} \leftarrow \text{CALCULATE\_ENERGY}(\pi, \Lambda_n, \mathcal{D})$
while OUTER-LOOP CRITERION SATISFIED do
    $n_{prev} \leftarrow n$
    $n \leftarrow \text{SAMPLE\_SIZE}(T^{(0)}, k, \sigma_{\max})$
    $\widehat{E}_{\Lambda_n,\pi} \leftarrow \text{ENERGY\_NORMALIZATION}(\widehat{E}_{\Lambda_n,\pi}, n_{prev}, n)$
    while INNER-LOOP CRITERION SATISFIED do
        $\pi_{cand} \leftarrow \text{GENERATE\_NEIGHBOR}(\pi)$
        $\Lambda'_n \leftarrow \text{TAKE\_SAMPLE}(\Lambda_N, n)$
        $\widehat{E}_{\Lambda'_n,\pi_{cand}} \leftarrow \text{CALCULATE\_ENERGY}(\pi_{cand}, \Lambda'_n, \mathcal{D})$
        $r \leftarrow \text{RAND}([0,1])$
        if ACCEPT($\widehat{E}_{\Lambda_n,\pi}$, $\widehat{E}_{\Lambda'_n,\pi_{cand}}$, $T^{(k)}$, $r$) then
            $\pi \leftarrow \pi_{cand}$
            $\widehat{E}_{\Lambda_n,\pi} \leftarrow \widehat{E}_{\Lambda'_n,\pi_{cand}}$
        end if
    end while
    $T^{(k+1)} \leftarrow \text{UPDATE\_TEMPERATURE}(T^{(k)})$
    $k \leftarrow k+1$
end while
return $\pi$
```

3. Application: DR pre-screening

DR is a complication of diabetes mellitus caused by progressive damage to the blood vessels in the retina, which is the light-sensitive lining in the back of the eye. DR is one of the leading causes of vision loss worldwide, but the risk of blindness can be significantly reduced through early diagnosis and timely treatment [12]. Therefore, patients with diabetes mellitus should undergo regular DR screening, but the manual grading of cases is resource-demanding and prone to human error. Consequently, over the last two decades, considerable efforts have been made to establish reliable automated methods to facilitate the mass screening of DR using color retinal photographs and various working principles, such as red and bright lesion detection [13,14], feature extraction and classification [15,16], and deep learning [17,18].

3.1. DR screening based on MA detection

Several of the methods mentioned above aim to assign grades to input retinal images according to the severity of DR. However, even the seemingly simpler problem of classifying retinal images into *healthy* and *diseased* categories is not yet considered to have been solved. Automatically selecting and prioritizing cases with a higher likelihood of disease could significantly facilitate the detection of DR in a mass screening scenario because only approximately 35% of patients with diabetes mellitus have DR [12].

MAs are tiny swellings in the blood vessels (see Fig. 2) and the earliest clinical signs of DR, where the number of MAs is strongly correlated with its severity [19]. Consequently, the accurate detection of MAs is crucially important for recognizing DR, especially in its early stage.

Several methods have been developed to directly screen for DR based on the presence of MAs. The method proposed by Hipwell et al. [20] is based on the results reported by Cree [21] and it can detect MAs using red-free retinal images. After removing variation in the background intensity, small round objects are extracted as candidates. Each MA candidate is then classified using intensity and size features. Fleming et al. [22] proposed a method that uses contrast normalization and vessel removal to improve MA detection, and they also evaluated their method for image classification. The method developed by Bhalerao et al. [23] is based on filtering using complex-valued circular-symmetric filters and morphological

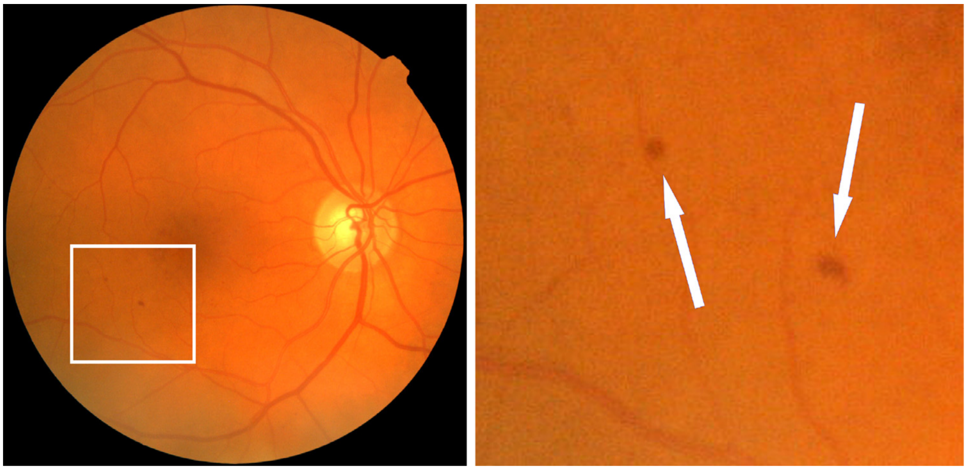


Fig. 2. MAs in a retinal image.

analysis of the candidate regions to reduce the false positive rate. In particular, they aimed to detect severe, sight-threatening DR. Giancardo et al. [24] proposed a method that discards the background areas, before calculating the Radon transform and extracting a feature vector, which is subsequently classified using principal component analysis and a nonlinear support vector machine. The results obtained by the methods mentioned above confirm that MA detection is a reasonable approach for DR pre-screening. For further details of the performance of MA-based DR classification methods, see Section 4.4.

A possible approach for further increasing the accuracy of MA detection involves creating an ensemble of detectors based on different working principles and models. To demonstrate the efficiency of the proposed method, in our case study application, we considered two ensembles for the binary classification of retinal images into *healthy* or *diseased* categories based solely on the presence of MAs. Next, we describe the members of our ensembles, the steps in the ensemble creation process, and the design choices required to implement the stochastic search method.

3.2. Ensemble creation method

We considered two MA detector ensembles with nine and ten members, respectively. The nine members of *Ensemble 1* were based on traditional object detector methods [3]. This ensemble was extended to *Ensemble 2* by adding one more detector based on the fusion of two deep convolutional neural networks (DCNNs).

The traditional MA detectors in our ensembles were formed as ⟨preprocessing method, candidate extractor⟩ pairs (⟨PP, CE⟩) as recommended in a previous study [25]. A ⟨PP, CE⟩ pair applied the PP to the input retinal image and the CE to its output; thus, a ⟨PP, CE⟩ pair extracted a set of MA candidates by acting as a single detector algorithm. The individual ⟨PP, CE⟩ detectors comprised the following components:

- PPs: Contrast limited adaptive histogram equalization (CLAHE) [26]; Illumination equalization (IE) [26]; Vessel removal with inpainting (VR) [27,28]; Walter-Klein (WK) [29]; No preprocessing (NP).
- CEs: Lázár et al. [30]; Walter et al. [31]; Zhang et al. [32].

To extend our former ⟨PP, CE⟩ ensemble [3] with a member based on deep neural networks, we employed the method proposed by Harangi et al. [33], which organizes two DCNNs into a single architecture by connecting them in a shared fully connected layer in order to recognize MAs in retinal images. The advantage of this approach is that the combined architecture can be trained as a single neural network, where the training of both DCNNs is affected by the predictions of each, thereby improving the detection accuracy. The input retinal image was divided into subimages to provide the required input for the combined DCNN. An input image was labeled as *diseased* if the presence of an MA was predicted in any of the subimages at a confidence level threshold of 0.5 to 0.95, depending on its parameter.

It should be noted that MAs are dot-like lesions (especially in lower resolution retinal images), so the MA detector components of our ensembles were implemented to extract the MA centers (i.e., the coordinates of a pixel) as candidates instead of image subregions.

Table 1 summarizes the members of the two ensembles used in our study. Ensemble 1 comprised nine MA detectors $D_1, \ldots, D_9$ with the indicated ⟨PP, CE⟩ pairs (see also [3]), and Ensemble 2 included an additional DCNN member D_{10}.

The detectors listed in Table 1 have various numbers of adjustable parameters. However, to make the optimization problem more tractable, we considered only that parameter for each detector that had the most significant effect on the output. In particular, the parameters $\pi_1, \ldots, \pi_4$ control thresholds for the scores assigned to the MA candidates, π_5 and π_6 control size thresholds for the diameter closing results, $\pi_7, \ldots, \pi_9$ control thresholds for the correlation map of the image used to extract candidates, and π_{10} controls the confidence threshold for MA candidates. The possible settings for each $\pi_i \in \Pi_i$ $(i = 1, \ldots, 10)$ are shown in Table 1. Overall, there are $20^4 \times 30^2 \times 10^3$ and $20^4 \times 30^2 \times 10^3 \times 6$ possible different parameter settings for Ensembles 1 and 2, respectively.

To fuse the MA candidates obtained by the individual detectors $D_1^{(\pi_1)}, \ldots, D_{10}^{(\pi_{10})}$ for a given image λ via $\mathcal{D}^{(\pi)}(\lambda) = \cup_{i=1}^{10} D_i^{(\pi_i)}(\lambda)$, we define a confidence measure to describe the rate of agreement by the members regarding the specific candidates. Thus, a proximity relation $\cong$ is introduced to decide whether or not two candidates indicate the same MA object. For the MA candidates c_1, c_2, we say that $c_1 \cong c_2$ if their Euclidean distance is smaller than a given threshold. Now, the confidence of the ensemble $conf_{\mathcal{D}^{(\pi)}}(c)$ regarding any of its candidates $c \in \mathcal{D}^{(\pi)}(\lambda)$ is defined as

$$conf_{\mathcal{D}^{(\pi)}}(c) = |\{D_i^{(\pi)} \in \mathcal{D}^{(\pi)} : \exists c' \in D_i^{(\pi)}(\lambda) : c \cong c'\}| / |\mathcal{D}^{(\pi)}|. \quad (16)$$

The ensemble candidates $\mathcal{D}^{(\pi)}(\lambda)$ are classified based on the degree of confidence for the subsequent labeling of the image. Accordingly, the α–level candidates of $\mathcal{D}^{(\pi)}$ are defined as

$$\left(\mathcal{D}^{(\pi)}(\lambda)\right)_\alpha = \{c \in \mathcal{D}^{(\pi)}(\lambda) \ : \ conf_{\mathcal{D}^{(\pi)}}(c) \geq \alpha\}, \quad (17)$$

where $1/|\mathcal{D}^{(\pi)}| \leq \alpha \leq 1$.

Table 1
Members of the ensembles.

		Comp.	PP	CE	Parameter domain
Ensemble 2	Ensemble 1	D_1	NP	Lázár *et al.*	$\Pi_1 = \{1, 2, \ldots, 20\}$
		D_2	CLAHE	Lázár *et al.*	$\Pi_2 = \{1, 2, \ldots, 20\}$
		D_3	IE	Lázár *et al.*	$\Pi_3 = \{1, 2, \ldots, 20\}$
		D_4	VR	Lázár *et al.*	$\Pi_4 = \{1, 2, \ldots, 20\}$
		D_5	NP	Walter *et al.*	$\Pi_5 = \{1, 2, \ldots, 30\}$
		D_6	CLAHE	Walter *et al.*	$\Pi_6 = \{1, 2, \ldots, 30\}$
		D_7	NP	Zhang *et al.*	$\Pi_7 = \{1, 2, \ldots, 10\}$
		D_8	VR	Zhang *et al.*	$\Pi_8 = \{1, 2, \ldots, 10\}$
		D_9	WK	Zhang *et al.*	$\Pi_9 = \{1, 2, \ldots, 10\}$
		D_{10}	NP	Harangi *et al.*	$\Pi_{10} = \{0, 1, \ldots, 5\}$

3.3. SA design choices

A number of design choices must be made to implement SA. In particular, we have to specify the method for generating the initial state, the neighborhood function, the acceptance criterion, the cooling schedule, and the termination criterion. We adjusted and implemented the corresponding components of Algorithm 1 as follows, using the line numbers for reference.

- *Input – Initial value of the control parameter* $T^{(0)}$: The initial temperature $T^{(0)}$ should be determined to allow virtually all state transitions to be accepted. Kirkpatrick et al. [6] suggested that a suitable value should result in an initial acceptance probability χ_0 of about 0.8. Thus, by using (19), we calculate $T^{(0)}$ as

  $$T^{(0)} = -\frac{\Delta E_{max}}{\ln(\chi_0)} = -\frac{1}{\ln(0.8)} \approx 4.5,$$

  where we note the maximal possible energy difference between any two states $\Delta E_{\max} = 1$ because the energy lies in the interval [0,1] in our case (see Section 4).
- *Line 2 – Initial state*: For each member of the ensemble, a valid parameter value is randomly selected to form an initial state.
- *Line 6 – Termination criterion* OUTER-LOOP CRITERION: When the temperature falls below the final value $T^{(k_{\max})}$, the search is stopped. At the final temperature $T^{(k_{\max})}$, the acceptance probability $\chi_{k_{\max}}$ should be almost 0. In a similar manner to the initial temperature, the final temperature is calculated as:

  $$T^{(k_{max})} = -\frac{\Delta E_{min}}{\ln(\chi_{k_{max}})} \quad \text{with} \quad \Delta E_{min} = \frac{N-1}{N}. \qquad (18)$$

  For example, if we set $\chi_{k_{\max}} = 10^{-1000}$ and consider a large population with $\frac{N-1}{N} \approx 1$, we obtain:

  $$T^{(k_{max})} = -\frac{1}{\ln(10^{-1000})} \approx 0.00043.$$
- *Line 10 – Thermal equilibrium criterion* INNER-LOOP CRITERION: This criterion is omitted in our implementation. The statements in the inner loop are executed once.
- *Line 11 – Neighborhood function* GENERATE_NEIGHBOR: We define a neighborhood with a size that decreases linearly in inverse proportion to the number of search iterations. For each parameter of the ensemble, a maximal distance is determined within which a new valid parameter value is randomly selected in each iteration. This distance is the length of the range of the parameter multiplied by (1 – the ratio of the index of the current search step and the maximum number of search steps).
- *Line 15 – Acceptance rule* ACCEPT: We employ the Metropolis acceptance criterion because of its widespread use and attractive properties [1]. The acceptance probability is calculated as:

  $$\chi_{\pi,\pi_{cand}} = \begin{cases} \exp\left(\frac{\widehat{E}_{\Lambda_n,\pi} - \widehat{E}_{\Lambda'_n,\pi_{cand}}}{T}\right), & \text{if } \widehat{E}_{\Lambda'_n,\pi_{cand}} > \widehat{E}_{\Lambda_n,\pi}, \\ 1, & \text{otherwise}, \end{cases} \qquad (19)$$

  where T is the current temperature, Λ_n and Λ'_n are two samples of size n, and $\widehat{E}_{\Lambda_n,\pi}$ and $\widehat{E}_{\Lambda'_n,\pi_{cand}}$ are the energy function values in the current and candidate states, respectively.
- *Line 20 – Annealing function* UPDATE_TEMPERATURE: We employ the exponential cooling schedule proposed in a previous study (10). α is determined to have exactly $k_{\max} = 1000$ iterations:

  $$\alpha = \left(\frac{T^{(k_{max})}}{T^{(0)}}\right)^{\frac{1}{k_{max}}} \approx 0.997.$$

4. Experimental results

In this section, we present the methods and results of our experiments. First, we describe the datasets employed, then discuss the assessment of the proposed method by performing parameter optimization of our ensembles for DR pre-screening and MA detection and provide the corresponding experimental results. Finally, we give some implementation details.

4.1. Datasets

Parameter optimization was performed for the ensembles using the publicly available dataset e-ophtha-MA [34] and the test part of the dataset provided by EyePACS for a DR grading competition held by Kaggle [35]. We will refer to the latter dataset as Kaggle EyePACS in the following. The contents of the two datasets are described as follows.

- *e-ophtha-MA*: The e-ophtha-MA dataset comprises 381 color retinal images with four different resolutions ranging from 1440 × 960 to 2544 × 1696 pixels, where 233 images depict healthy retinas (R0 class) and 148 images show various severity levels of DR (R1–R4 classes) containing a total of 1306 MAs. We used this dataset mainly because it contains precise MA ground truth data for the images.
- *Kaggle EyePACS*: The Kaggle EyePACS dataset comprises 35 126 color retinal images with various resolutions ranging from 400 × 315 to 5184 × 3456 pixels, where 25 810 images are labeled as healthy (R0), 2443 as mild DR (R1), 5292 as moderate DR (R2), 873 as severe DR (R3), and 708 as proliferative DR (R4). The images in this dataset were acquired under various imaging conditions using different models and types of cameras. Furthermore, as stated in the dataset description [35],

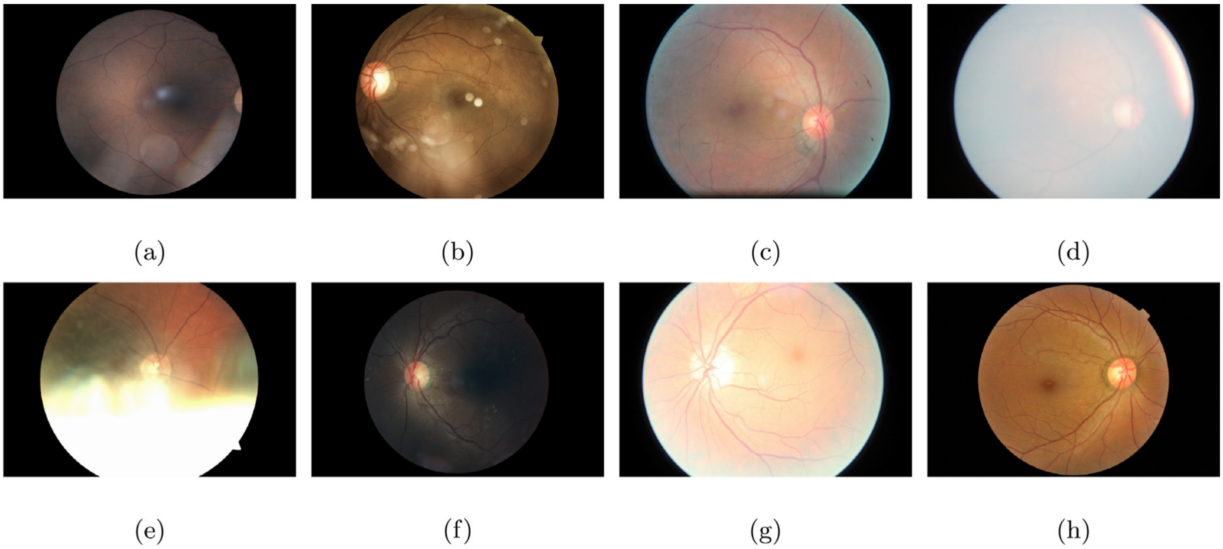


Fig. 3. Sample images from the Kaggle EyePACS dataset showing typical artifacts and imaging errors: (a) camera artifacts, (b) lens condensation, (c) dust, (d) blur, (e) reflection, (f) underexposure, (g) overexposure, and (h) no artifacts.

Table 2
Contents of the datasets.

	Subset	Healthy	Diseased					Total
		R0	R1	R2	R3	R4	R1-R4	
e-ophtha-MA		148	-	-	-	-	233	381
	training	100	-	-	-	-	100	200
	test	48	-	-	-	-	48	96
	not used	-	-	-	-	-	85	85
Kaggle EyePACS		25,810	2443	5292	873	708	9316	35,126
	training	6211	1629	3528	582	472	6211	12,422
	test	3105	814	1764	291	236	3105	6210
	not used	16,494	-	-	-	-	-	16,494

some images are labeled incorrectly, affected by artifacts, out of focus, underexposed, or overexposed (see Fig. 3). According to previous studies using this dataset (e.g., see [36]) approximately 20–30% of the images are of poor quality or have incorrectly assigned labels. We used this dataset mainly because to the best of our knowledge, this is the largest freely available dataset that contains DR severity label ground truth data for the images.

Despite the known issues with Kaggle EyePACS, we used the images from this dataset as provided and did not perform any resource-demanding data cleaning steps (e.g., manually filtering the gradable images) because our main aim was to show that the proposed evaluation method can preserve the achievable solution quality while reducing the runtime. Clearly, due to the high proportion of poor quality or incorrectly labeled images, a lower diagnostic efficiency can be expected for Kaggle EyePACS than e-ophtha-MA using either the standard SA or the proposed method.

The contents of the datasets used in the experiments described in Sections 4.2 and 4.3 are summarized in Table 2 and Fig. 4.

4.2. DR pre-screening

For DR pre-screening, the aim of the optimization process was to find the parameter setting π that maximized the performance of the ensemble $\mathcal{D}^{(\pi)}$ in terms of the diagnostic efficiency, i.e., maximizing the proportion of correctly classified images.

The output of the ensemble was a Bernoulli distributed random variable, where $X_{\mathcal{D}^{(\pi)}} = 1$ for correct classification and $X_{\mathcal{D}^{(\pi)}} = 0$ for incorrect classification. We considered that an image λ was classified correctly if it was annotated as positive in the ground truth and $\left|\left(\mathcal{D}^{(\pi)}(\lambda)\right)_\alpha\right| \geq 1$ (true positive), or annotated as negative and $\left|\left(\mathcal{D}^{(\pi)}(\lambda)\right)_\alpha\right| = 0$ (true negative). By contrast, λ was classified incorrectly if it was annotated as positive in the ground truth and $\left|\left(\mathcal{D}^{(\pi)}(\lambda)\right)_\alpha\right| = 0$ (false negative case), or annotated as negative and $\left|\left(\mathcal{D}^{(\pi)}(\lambda)\right)_\alpha\right| \geq 1$ (false positive case). The candidates for the ensembles were extracted at the confidence level of $\alpha = 0.5$, i.e., we used simple majority voting for this aim. The ensembles considered that an image was diseased if at least one MA was detected.

To optimize the DR pre-screening performance of the ensembles, we used the energy function estimate $\widehat{E}_{\Lambda_n}$ given in (4) corresponding to this implementation. It should be noted that in this case, the energy function is equivalent to the accuracy (*ACC*) measure given as

$$ACC = \frac{\text{number of true hits}}{\text{number of all images}} = \frac{TP + TN}{TP + TN + FP + FN}, \tag{20}$$

where *TP*, *TN*, *FP*, and *FN* are the numbers of true positive, true negative, false positive, and false negative hits, respectively. Furthermore, we calculated the sensitivity (*SE*) and specificity (*SP*) measures as:

$$SE = \frac{TP}{TP + FN}, \quad SP = \frac{TN}{TN + FP}. \tag{21}$$

For further details of *ACC*, *SE*, and *SP*, please refer to [37].

To evaluate the proposed method, we conducted 10-times cross-validation with repeated random subsampling of the datasets. For each round of the cross-validation process, we created new training and test subsets from the datasets described in

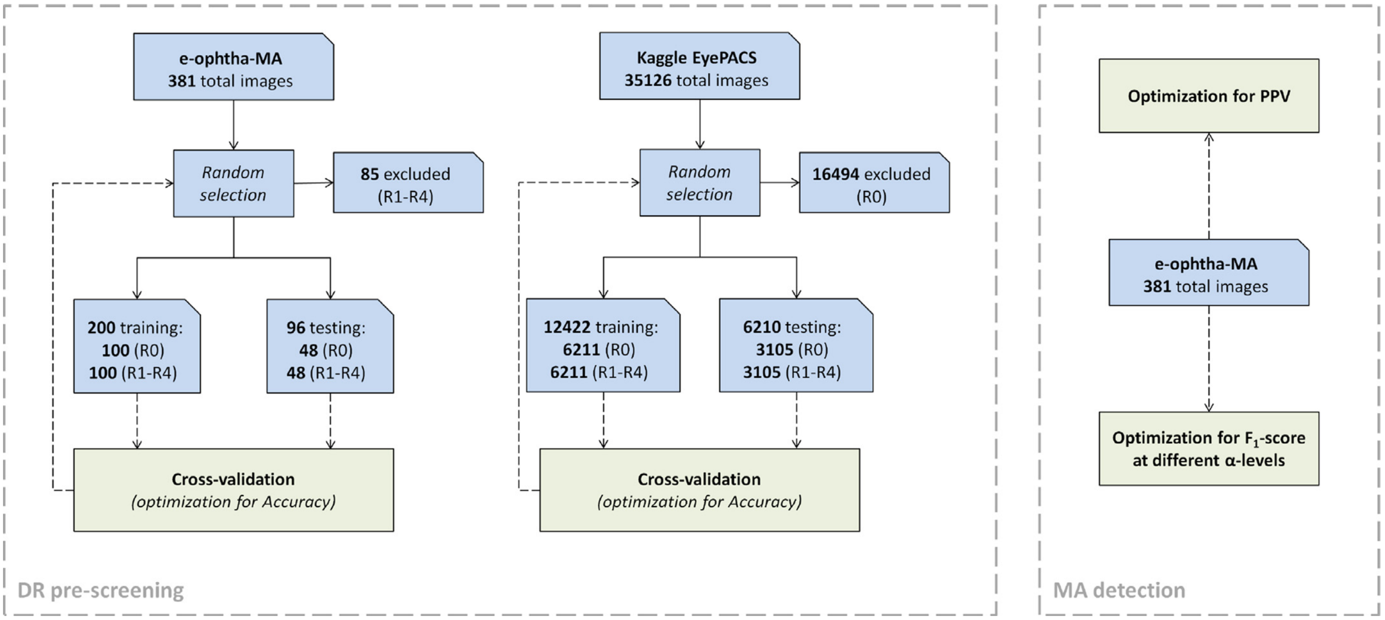

e-ophtha-MA
381 total images
Random selection
85 excluded (R1-R4)
200 training: 100 (R0) 100 (R1-R4)
96 testing: 48 (R0) 48 (R1-R4)
Cross-validation (optimization for Accuracy)
Kaggle EyePACS
35126 total images
Random selection
16494 excluded (R0)
12422 training: 6211 (R0) 6211 (R1-R4)
6210 testing: 3105 (R0) 3105 (R1-R4)
Cross-validation (optimization for Accuracy)
DR pre-screening
Optimization for PPV
e-ophtha-MA
381 total images
Optimization for F1-score at different α-levels
MA detection

Fig. 4. Visual overview of the datasets and the main evaluation approaches used in our experiments.

Table 3
DR pre-screening – Results of the 10-times cross-validation using the e-ophtha-MA dataset.

		Subset	*ACC*	*SE*	*SP*	*t*
Ensemble 1	SA	training	0.862 (± 0.014)	0.831 (± 0.027)	0.893 (± 0.021)	773.6 (± 100.5)
		test	0.8125 (± 0.0339)	0.7625 (± 0.0486)	0.8625 (± 0.0529)	-
	SA-SBE	training	0.858 (± 0.0127)	0.847 (± 0.019)	0.869 (± 0.0342)	187.3 (± 53.1)
		test	0.8115 (± 0.0347)	0.7833 (± 0.0458)	0.8396 (± 0.0618)	-
Ensemble 2	SA	training	0.8925 (± 0.014)	0.883 (± 0.029)	0.902 (± 0.0426)	1586.5 (± 189.7)
		test	0.8448 (± 0.0359)	0.825 (± 0.0792)	0.8647 (± 0.0545)	-
	SA-SBE	training	0.896 (± 0.0089)	0.889 (± 0.0262)	0.903 (± 0.0329)	591.4 (± 151.8)
		test	0.8813 (± 0.0256)	0.8833 (± 0.0468)	0.8791 (± 0.0445)	-

Section 4.1, with a training to test ratio of approximately 2:1 (see Fig. 4). In the case of e-ophtha-MA, 85 randomly selected images from the R1–R4 classes in the dataset were excluded from each round in order to ensure that we had the same amount of images in the R0 and R1–R4 classes. Next, 100 images were randomly selected from each of the R0 and R1–R4 classes for the training subset and the remaining 48 in each were used for testing. In the case of Kaggle EyePACS, 16,494 randomly selected images from the R0 class in the dataset were excluded in each round for the same reason explained above for e-ophtha-MA. Next, 6211, 1629, 3528, 582, and 472 images were randomly selected from the R0, R1, R2, R3, and R4 classes, respectively, for the training subset and the remaining 3105, 814, 1764, 291, and 236 images were used for testing.

The optimal parameter settings obtained in each round of the cross-validation process using a training subset were evaluated using the corresponding test subset.

The main results obtained in these experiments are summarized in Tables 3 and 4. In these tables, we present the average *ACC*, average *SE*, and average *SP* values, as well as the average runtimes *t* (in seconds) and the corresponding standard deviations calculated based on the results of the 10-times cross-validation using the e-ophtha-MA and the Kaggle EyePACS datasets, respectively. The runtimes for the test subsets are omitted because only single evaluations were needed.

Tables 3 and 4 clearly suggest that the proposed method preserved the quality of the solution obtained using the standard SA but with significantly lower time requirements. In addition, SA-SBE exhibited stable behavior in terms of the standard deviations of *ACC*, *SE*, and *SP*, and also when compared to the standard SA. It should be noted that the average *ACC* was lower using Kaggle EyePACS than e-ophtha-MA because of the artifact issues discussed in Section 4.1. However, there were no significant differences between the average *ACC* values obtained with the two optimization methods. The differences in the performance of SA and SA-SBE are also highlighted in Table 5.

We also checked the contribution of the DCNN member to the ensemble. Table 6 shows the individual performance of the DCNN approach together with those of the ensembles using the results obtained from the 10-times cross-validation with SA-SBE. With e-ophtha-MA, the individual performance of the DCNN was higher than that of the traditional image processing-based Ensemble 1. However, their combined performance (Ensemble 2) was better, especially considering the more balanced *SE* and *SP* values. With Kaggle EyePACS, the DCNN component still performed better than Ensemble 1, and Ensemble 2 obtained the highest performance with an improvement in *SP*, although the performance gain was less remarkable with this dataset.

4.3. MA detection

In Section 4.2, we presented evaluations of our sampling-based search strategy via the optimization of our ensembles for DR pre-screening. Next, we demonstrate that the same ensembles can also be optimized using our approach for the accurate detection of MAs. We used the whole e-ophtha-MA dataset in these experiments.

The α–level candidates of an ensemble extracted using the parameter setting π for an image λ $\left(\mathcal{D}^{(\pi)}(\lambda)\right)_{\alpha}$ were compared with

Table 4
DR pre-screening – Results of the 10-times cross-validation using the Kaggle EyePACS dataset.

		Subset	*ACC*	*SE*	*SP*	*t*
Ensemble 1	SA	training	0.6516 (± 0.0047)	0.5697 (± 0.022)	0.7336 (± 0.0243)	11,936.2 (± 932.9)
		test	0.6441 (± 0.0125)	0.5622 (± 0.0319)	0.726 (± 0.0249)	-
	SA-SBE	training	0.6488 (± 0.0064)	0.5643 (± 0.0249)	0.7334 (± 0.0299)	1685.4 (± 710)
		test	0.6396 (± 0.0041)	0.5556 (± 0.0282)	0.7236 (± 0.0314)	-
Ensemble 2	SA	training	0.6701 (± 0.0068)	0.5556 (± 0.0307)	0.7846 (± 0.0251)	87,198.2 (± 9111.4)
		test	0.6649 (± 0.0079)	0.5511 (± 0.0250)	0.7787 (± 0.0215)	-
	SA-SBE	training	0.6672 (± 0.0074)	0.5476 (± 0.0212)	0.7869 (± 0.0216)	9611.4 (± 3567.2)
		test	0.6580 (± 0.006)	0.5415 (± 0.0282)	0.7745 (± 0.0231)	-

Table 5
Comparison of SA and SA-SBE in terms of the average solution quality and runtime based on 10-times cross-validation.

		e-ophtha-MA (training)		Kaggle EyePACS (training)	
		ACC	*t*	*ACC*	*t*
Ensemble 1	SA	0.862	773.6	0.6516	11,936.2
	SA-SBE	0.858	187.3	0.6488	1685.4
	Difference	−0.004 (−0.46%)	−586.3 (−75.79%)	−0.0028 (−0.43%)	−10,250.8 (−85.88%)
Ensemble 2	SA	0.8925	1586.5	0.6701	87,198.2
	SA-SBE	0.896	591.4	0.6672	9611.4
	Difference	0.0035 (0.39%)	−995.1 (−62.72%)	−0.0029 (−0.43%)	−77,586.8 (−88.98%)

Table 6
Comparison of the DR pre-screening performance of the ensembles and the DCNN member.

	e-ophtha-MA (test)			Kaggle EyePACS (test)		
	ACC	*SE*	*SP*	*ACC*	*SE*	*SP*
Ensemble 1	0.8115	0.7833	0.8396	0.6396	0.5556	0.7236
DCNN (D_{10})	0.8427	0.7458	0.9396	0.6536	0.6577	0.6496
Ensemble 2	0.8813	0.8833	0.8791	0.6580	0.5415	0.7745

Table 7
MA detection performance of the ensembles using the e-ophtha-MA dataset.

	Ensemble 1		Ensemble 2	
	$\overline{PPV}$	*t*	$\overline{PPV}$	*t*
SA	0.9921	1451	0.9974	6743
SA-SBE	0.9895	172	0.9974	238

a set of MA centers (which were extracted from the ground truth masks provided for the image) using a method similar to that described for the fusion of MA candidates in Section 3.2. If the Euclidean distance of the centers of a candidate and a manually annotated MA center was smaller than a given threshold, it was considered a true positive, otherwise it was treated as a false positive. Furthermore, each missed annotated MA was considered a false negative. The threshold was set to 5 pixels for our experiments, where this value was selected according to the average MA size in the images.

First, we optimized the parameter settings for our ensembles to maximize the mean positive predictive value ($\overline{PPV}$) (see [37]) over a set of n images, i.e., the average percentage of true MAs in the output of the detector ensemble:

$$\overline{PPV} = \frac{1}{n}\sum_{i=1}^{n}\frac{TP_{\lambda_i}}{TP_{\lambda_i}+FP_{\lambda_i}}, \tag{22}$$

where λ_i is the ith image, and TP_{λ_i} and FP_{λ_i} are the numbers of true positive and false positive MA candidates, respectively, in the output of the ensemble for the image λ_i.

We repeated the parameter optimization process four times with both ensembles. Table 7 shows the best lesion-level performance obtained with Ensemble 1 and Ensemble 2 for $\overline{PPV}$ at α–level = 0.5. Our conclusion based on these results is similar to that for the image-level results where significant reductions in the computational time were achieved with SA-SBE while the quality of solution obtained with the standard SA was preserved.

$\overline{PPV}$ is useful for optimizing our ensembles for a DR pre-screening approach based solely on the presence of MAs because a low number of false positives is a desirable characteristic of this type of system. However, $\overline{PPV}$ only considers the ratio of the number of true positives relative to the number of all positives, whereas the number of false negatives is ignored. Thus, if the ensemble finds some true positives in each image and no false positives, then $\overline{PPV}$ is 1, even if the ensemble misses numerous MAs in the images. Therefore, it would be misleading to use $\overline{PPV}$ only to assess the MA detection performance of the ensembles.

Thus, we also performed optimization for the mean F_1–score ($\overline{F_1}$) over a set of n images. $\overline{F_1}$ was considered an appropriate measure for our study because it is the average harmonic mean of *PPV* and *SE* calculated as

$$\overline{F_1} = \frac{1}{n}\sum_{i=1}^{n}\frac{2\ TP_{\lambda_i}}{2\ TP_{\lambda_i}+FP_{\lambda_i}+FN_{\lambda_i}}, \tag{23}$$

where the previously defined notations apply and FN_{λ_i} denotes the number of false negative MA candidates on λ_i. Fig. 5 shows examples of true positive, false positive, and false negative MA candidates.

Based on the optimization results obtained for $\overline{F_1}$, Fig. 6 shows the respective free-response receiver operating characteristic (FROC) curves [38] for Ensemble 1 and Ensemble 2, where *SE* is plotted against the average number of false positives per image (*FPI*). To measure the *SE* at different average *FPI* levels, we looped the α–level confidence value of the ensembles from 0.1 to 1 with a step size of 0.1 and repeated the optimization process accordingly. The higher performance of Ensemble 2 compared with Ensemble 1 is clearly visible in Fig. 6.

4.4. DR classification at different confidence levels

In an additional experiment, we evaluated the DR classification performance of our ensembles at different confidence levels. In this experiment, we repeated the parameter optimization process four

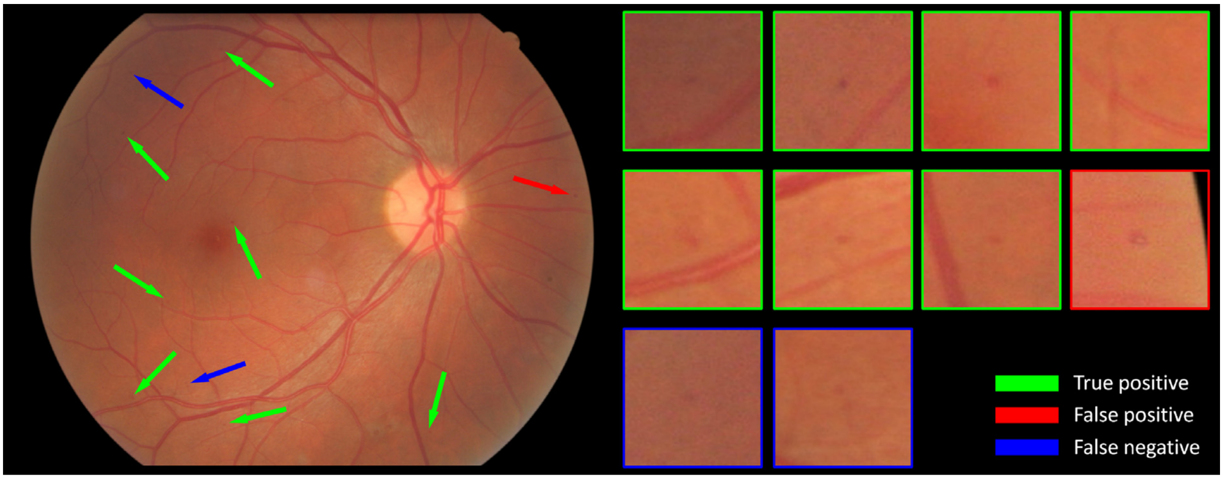


Fig. 5. Examples of true positive, false positive, and false negative MA candidates found in an image from the e-ophtha-MA dataset by Ensemble 2.

Table 8
DR classification performance of the ensembles at different α–levels using the e-ophtha-MA dataset.

	α	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
Ensemble 1	*SE*	1	1	1	0.9527	0.7838	0.4122	0.0203	0.0203	0	0
	SP	0	0	0.0558	0.4206	0.9013	0.9828	1	1	1	1
	ACC	0.3885	0.3885	0.4226	0.6273	0.8556	0.7612	0.6194	0.6194	0.6115	0.6115
Ensemble 2	*SE*	1	1	1	0.9662	0.8986	0.4932	0.0743	0.0068	0	0
	SP	0	0.0178	0.2103	0.5751	0.9270	1	1	1	1	1
	ACC	0.3885	0.3990	0.5170	0.7270	0.9160	0.8031	0.6404	0.6141	0.6115	0.6115

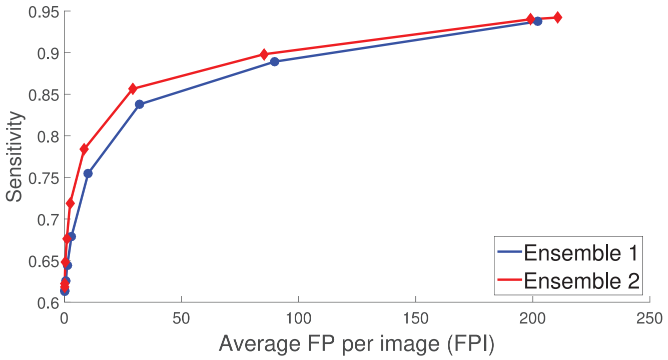


Fig. 6. MA detection performance – FROC curves obtained for Ensemble 1 (blue) and Ensemble 2 (red). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

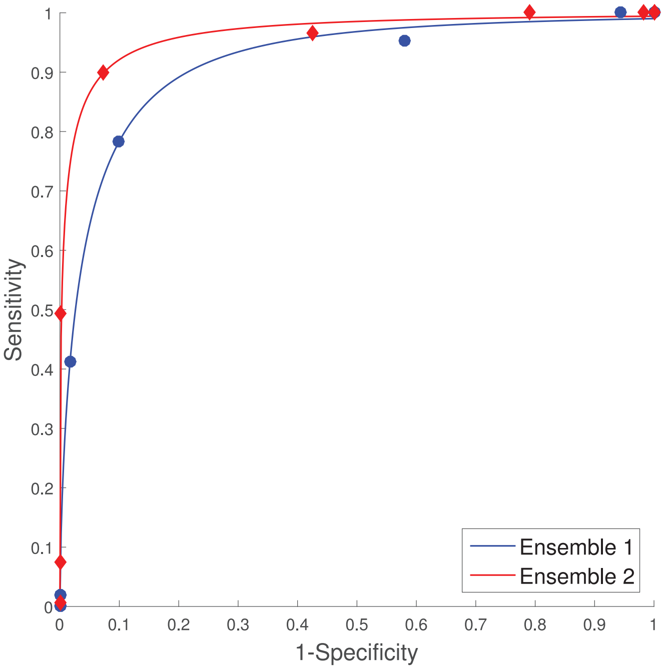


Fig. 7. DR classification performance – ROC curves obtained for Ensemble 1 (blue) and Ensemble 2 (red) using the e-ophtha-MA dataset. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

times with both ensembles for *ACC* using the whole e-ophtha-MA dataset and α–level = 0.5. Using the parameter setting with the highest *ACC* value in the four tests, we measured *ACC*, *SE*, and *SP* at α–levels ranging from 0.1 to 1 with a step size of 0.1. The corresponding results are provided in Table 8. Furthermore, the fitted receiver operating characteristic (ROC) curves obtained for the ensembles are presented in Fig. 7, which again showed that Ensemble 2 performed better than Ensemble 1.

Finally, Table 9 gives the DR classification performance of Ensemble 2 at α–level = 0.5 and those of the methods described in Section 3.1. The reported performance levels are not directly comparable because of the different datasets and evaluation methods employed, but it can be observed that the performance of Ensemble 2 is competitive in this field.

4.5. Implementation and hardware details

SA-SBE was implemented in Java SE 8 and also used for the SA tests with sampling disabled. All the detector outputs were stored in memory during the search and the evaluation of the energy function was parallelized at the image level in order to reduce the time required to find a solution. The reported runtimes exclude the time required for loading the input files and other overheads. The results with the e-ophtha-MA dataset were acquired using a computer equipped with two 6-core AMD Opteron 2423 HE processors and 32 GB DDR2 RAM. The results with the Kaggle Eye-

Table 9
Performance of MA-based DR classification methods.

Method	Performance	Dataset used
Hipwell et al. [20]	*SE*: 0.78, *SP*: 0.91	non-public (3783 images, 956 with DR)
Fleming et al. [22]	*SE*: 0.854, *SP*: 0.831	non-public (1441 images, 356 with DR)
Bhalerao et al. [23]	*SE*: 0.826, *SP*: 0.802	DIARETDB1 (89 images, 80 with DR)
Giancardo et al. [24]	*AUC*: 0.854	Messidor (1200 images, 654 with DR)
Ensemble 2	*SE*: 0.899, *SP*: 0.927 (*AUC*: 0.965)	e-ophtha-MA (381 images, 233 with DR)

PACS dataset were acquired using two computers, where each was equipped with a 4-core Intel Xeon W-2123 processor and 64 GB DDR4 RAM.

5. Conclusions

In object detection applications, it is common to optimize systems using objective functions that are calculated as an average over a dataset. Our motivation for constructing the proposed method was to provide a theoretically established way for reducing the time required for optimization but without any significant loss of the solution quality if the dataset considered is large. In Section 2, we proposed a sampling strategy to ensure that SA exhibits the same convergence in probability using sampling-based evaluation as that using complete evaluation. Our experimental results in Section 4 demonstrated that SA-SBE can provide the same solution quality as SA for our parameter optimization problems. The proposed evaluation method is domain independent and easy to adapt to problems where evaluation over large datasets is required. Our method does not incorporate complex techniques for the determination of the required sample size (e.g., monitoring changes in energy) or sample selection (e.g., finding the critical samples in classes) to accelerate the search process.

For practical problems, it is typically possible to empirically determine a fixed sampling rate for the evaluation in SA in order to obtain solutions with adequate quality and reduce the runtime. However, using the same sample size in each iteration would not necessarily provide the same solution quality as a complete evaluation. In the case of SA, according to (1), the standard deviation of the energy noise must approach 0 faster than the temperature to maintain the convergence in probability, i.e., the sampling rate must approach 1 faster in our case. Clearly, for any fixed sample size $n_{const} < N$, there is a temperature level $T^{(l)}$ $(0 \leq l < k_{\max})$ up to n_{const} would be larger than the minimum sample size required to maintain the convergence in probability, and thus the search would be slower than possible, and after reaching $T^{(l)}$, samples of size n_{const} will be insufficient and the search convergence will deteriorate, thereby potentially decreasing the performance.

In future research, we plan to investigate embedding sampling-based evaluation with a dynamic sample size in other stochastic search methods. In addition, we plan to generalize the proposed dataset-level sampling strategy to systematic image-level sampling, i.e., to image downsampling, where estimation of the noise originating from the sampling is required.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

This work has been supported by the project EFOP-3.6.2-16-2017-00015 supported by the European Union, co-financed by the European Social Fund.

References

[1] D. Delahaye, S. Chaimatanan, M. Mongeau, Simulated annealing: from basics to applications, in: Handbook of Metaheuristics, Springer International Publishing, 2018, pp. 1–35, doi:10.1007/978-3-319-91086-4_1.

[2] M. Mohandes, M. Deriche, S.O. Aliyu, Classifiers combination techniques: a comprehensive review, IEEE Access 6 (2018) 19626–19639, doi:10.1109/access.2018.2813079.

[3] J. Tóth, L. Szakács, A. Hajdu, Finding the optimal parameter setting for an ensemble-based lesion detector, in: 2014 IEEE International Conference on Image Processing (ICIP), 2014, pp. 3532–3536, doi:10.1109/ICIP.2014.7025717.

[4] C.C. Aggarwal, Neural Networks and Deep Learning, Springer International Publishing, 2018, doi:10.1007/978-3-319-94463-0.

[5] S.B. Gelfand, S.K. Mitter, Simulated annealing with noisy or imprecise energy measurements, J. Optim. Theory Appl. 62 (1) (1989) 49–62, doi:10.1007/BF00939629.

[6] S. Kirkpatrick, C.D. Gelatt, M.P. Vecchi, Optimization by simulated annealing, Science 220 (4598) (1983) 671–680, doi:10.1126/science.220.4598.671.

[7] V. Černý, Thermodynamical approach to the traveling salesman problem: an efficient simulation algorithm, J. Optim. Theory Appl. 45 (1) (1985) 41–51, doi:10.1007/BF00940812.

[8] H.J. Kushner, Asymptotic global behavior for stochastic approximation and diffusions with slowly decreasing noise effects: global minimization via monte carlo, SIAM J. Appl. Math. 47 (1) (1987) 169–185, doi:10.1137/0147010.

[9] T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein, Introduction to Algorithms, third ed., The MIT Press, 2009.

[10] W.J. Gutjahr, G.C. Pflug, Simulated annealing for noisy cost functions, J. Global Optim. 8 (1) (1996) 1–13, doi:10.1007/bf00229298.

[11] R. Bethea, Statistical Methods for Engineers and Scientists, third ed., CRC Press, 2019.

[12] J.B. Jonas, C. Sabanayagam, Epidemiology and risk factors for diabetic retinopathy, in: Frontiers in Diabetes, S. Karger AG, 2019, pp. 20–37, doi:10.1159/000486262.

[13] B. Antal, A. Hajdu, An ensemble-based system for automatic screening of diabetic retinopathy, Knowl. Based Syst. 60 (2014) 20–27, doi:10.1016/j.knosys.2013.12.023.

[14] R. Biyani, B. Patre, Algorithms for red lesion detection in diabetic retinopathy: a review, Biomed. Pharmacother. 107 (2018) 681–688, doi:10.1016/j.biopha.2018.07.175.

[15] S. Morales, K. Engan, V. Naranjo, A. Colomer, Retinal disease screening through local binary patterns, IEEE J. Biomed. Health Inform. 21 (1) (2017) 184–192, doi:10.1109/jbhi.2015.2490798.

[16] A. Colomer, J. Igual, V. Naranjo, Detection of early signs of diabetic retinopathy based on textural and morphological information in fundus images, Sensors 20 (4) (2020) 1005, doi:10.3390/s20041005.

[17] K. Shankar, A.R.W. Sait, D. Gupta, S. Lakshmanaprabu, A. Khanna, H.M. Pandey, Automated detection and classification of fundus diabetic retinopathy images using synergic deep learning model, Pattern Recognit. Lett. 133 (2020) 210–216, doi:10.1016/j.patrec.2020.02.026.

[18] T. Araújo, G. Aresta, L. Mendonça, S. Penas, C. Maia, A. Carneiro, A.M. Mendonça, A. Campilho, DR|GRADUATE: Uncertainty-aware deep learning-based diabetic retinopathy grading in eye fundus images, Med. Image Anal. 63 (2020) 101715, doi:10.1016/j.media.2020.101715.

[19] J. Xu, X. Zhang, H. Chen, J. Li, J. Zhang, L. Shao, G. Wang, Automatic analysis of microaneurysms turnover to diagnose the progression of diabetic retinopathy, IEEE Access 6 (2018) 9632–9642, doi:10.1109/access.2018.2808160.

[20] J.H. Hipwell, F. Strachan, J.A. Olson, K.C. McHardy, P.F. Sharp, J.V. Forrester, Automated detection of microaneurysms in digital red-free photographs: a diabetic retinopathy screening tool, Diabet. Med. 17 (8) (2000) 588–594, doi:10.1046/j.1464-5491.2000.00338.x.

[21] M.J. Cree, J.A. Olson, K.C. McHardy, P.F. Sharp, J.V. Forrester, A fully automated comparative microaneurysm digital detection system, Eye 11 (5) (1997) 622–628, doi:10.1038/eye.1997.166.

[22] A.D. Fleming, S. Philip, K.A. Goatman, J.A. Olson, P.F. Sharp, Automated microaneurysm detection using local contrast normalization and local vessel detection, IEEE Trans. Med. Imaging 25 (9) (2006) 1223–1232, doi:10.1109/TMI.2006.879953.

[23] A. Bhalerao, A. Patanaik, S. Anand, P. Saravanan, Robust detection of microaneurysms for sight threatening retinopathy screening, in: 2008 Sixth Indian Conference on Computer Vision, Graphics Image Processing, 2008, pp. 520–527, doi:10.1109/ICVGIP.2008.25.

[24] L. Giancardo, T.P. Karnowski, K.W. Tobin, F. Meriaudeau, E. Chaum, Validation of microaneurysm-based diabetic retinopathy screening across retina fundus datasets, in: Proceedings of the 26th IEEE International Symposium on Computer-Based Medical Systems, 2013, pp. 125–130, doi:10.1109/CBMS.2013.6627776.

[25] B. Antal, A. Hajdu, An ensemble-based system for microaneurysm detection and diabetic retinopathy grading, IEEE Trans. Biomed. Eng. 59 (6) (2012) 1720–1726, doi:10.1109/TBME.2012.2193126.

[26] B. Harangi, A. Hajdu, Exudate detection in fundus images using active contour methods and region-wise classification, in: Biomedical Image Segmentation: Advances and Trends, CRC Press, 2019, pp. 157–186.

[27] S. Ravishankar, A. Jain, A. Mittal, Automated feature extraction for early detection of diabetic retinopathy in fundus images, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2009, pp. 210–217, doi:10.1109/CVPR.2009.5206763.

[28] P. Buyssens, M. Daisy, D. Tschumperle, O. Lezoray, Exemplar-based inpainting: technical review and new heuristics for better geometric reconstructions, IEEE Trans. Image Process. 24 (6) (2015) 1809–1824, doi:10.1109/tip.2015.2411437.

[29] T. Walter, J.C. Klein, Automatic detection of microaneurysms in color fundus images of the human retina by means of the bounding box closing, in: A. Colosimo, P. Sirabella, A. Giuliani (Eds.), Medical Data Analysis, Springer Berlin Heidelberg, Berlin, Heidelberg, 2002, pp. 210–220.

[30] I. Lázár, A. Hajdu, Microaneurysm detection in retinal images using a rotating cross-section based model, in: 2011 IEEE International Symposium on Biomedical Imaging: From Nano to Macro, 2011, pp. 1405–1409, doi:10.1109/ISBI.2011.5872663.

[31] T. Walter, P. Massin, A. Erginay, R. Ordonez, C. Jeulin, J.C. Klein, Automatic detection of microaneurysms in color fundus images, Med. Image Anal. 11 (6) (2007) 555–566, doi:10.1016/j.media.2007.05.001.

[32] B. Zhang, X. Wu, J. You, Q. Li, F. Karray, Detection of microaneurysms using multi-scale correlation coefficients, Pattern Recognit. 43 (6) (2010) 2237–2248, doi:10.1016/j.patcog.2009.12.017.

[33] B. Harangi, J. Tóth, A. Hajdu, Fusion of deep convolutional neural networks for microaneurysm detection in color fundus images, in: 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), 2018, pp. 3705–3708, doi:10.1109/EMBC.2018.8513035.

[34] E. Decenciére, G. Cazuguel, X. Zhang, G. Thibault, J.-C. Klein, F. Meyer, B. Marcotegui, G. Quellec, M. Lamard, R. Danno, D. Elie, P. Massin, Z. Viktor, A. Erginay, B. Laÿ, A. Chabouis, Teleophta: Machine learning and image processing methods for teleophthalmology, IRBM 34 (2) (2013) 196–203, doi:10.1016/j.irbm.2013.01.010. Special issue: ANR TECSAN: Technologies for Health and Autonomy

[35] Kaggle Inc., Diabetic Retinopathy Detection. https://www.kaggle.com/c/diabetic-retinopathy-detection, accessed: 2020-01-19.

[36] R.J. Chalakkal, W.H. Abdulla, S.S. Thulaseedharan, Quality and content analysis of fundus images using deep learning, Comput. Biol. Med. 108 (2019) 317–331, doi:10.1016/j.compbiomed.2019.03.019.

[37] P. Eusebi, Diagnostic accuracy measures, Cerebrovasc. Dis. 36 (4) (2013) 267–272, doi:10.1159/000353863.

[38] D. Chakraborty, Observer performance methods for diagnostic imaging: foundations, Modeling, and Applications with R-based Examples, CRC Press, 2017.

János Tóth received the M.Sc. degree in computer science from the University of Debrecen, Debrecen, Hungary, in 2010. Since 2018, he has been an Assistant Lecturer with the Department of Computer Graphics and Image Processing, Faculty of Informatics, University of Debrecen. He is a member of the IEEE, John von Neumann Computer Society, and the Hungarian Association for Image Analysis and Pattern Recognition.

Henrietta Tomán received the M.Sc. degree in mathematics in 2000 and the Ph.D. degree in 2015 from the University of Debrecen, Debrecen, Hungary. Since 2001, she has been an Assistant Lecturer, since 2009, she has been an Assistant Professor with the Department of Computer Graphics and Image Processing, Faculty of Informatics, University of Debrecen.

András Hajdu received the M.Sc. degree in mathematics from the Lajos Kossuth University, Debrecen, Hungary, in 1996, and the Ph.D. degree in mathematics and computer science from the University of Debrecen, Debrecen, in 2003. In 2017, the Hungarian Academy of Sciences awarded him the title Doctor of the Academy. Since 2001, he has been an Assistant Lecturer, since 2003, he has been an Assistant Professor, since 2009, he has been an Associate Professor, and since 2017, he has been a Full Professor with the Faculty of Informatics, University of Debrecen. Since 2011, he has been the Head of Department of Computer Graphics and Image Processing, Faculty of Informatics, University of Debrecen, and since 2019, he has been the Dean of the Faculty of Informatics, University of Debrecen. He is a member of the IEEE, János Bolyai Mathematical Society, John von Neumann Computer Society, Public Body of the Hungarian Academy of Sciences, and a Steering Committee Member of the Hungarian Association for Image Analysis and Pattern Recognition.