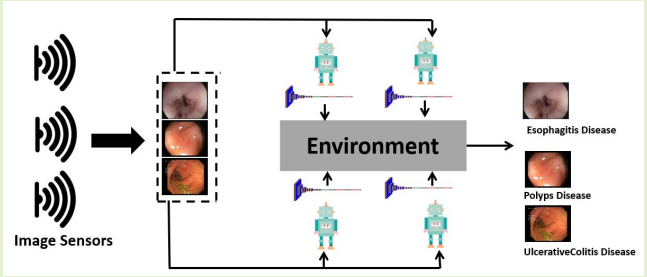
An Intelligent Collaborative Image-Sensing System for Disease Detection

Youcef Djenouri, Asma Belhadi, Anis Yazidi, Gautam Srivastava, *Senior Member, IEEE*, Pushpita Chatterjee, and Jerry Chun-Wei Lin, *Senior Member, IEEE*

***Abstract*—With the growth of smart medical devices and applications in smart hospitals, home care facilities, nursing, and the Internet of Medical Things (IoMT) are becoming more ubiquitous. It uses smart medical devices and cloud computing services, and basic Internet of Things (IoT) technology, to detect key body indicators, monitor health situations, and generate multivariate data to provide just-in-time healthcare services. In this article, we present a novel collaborative disease detection system based on IoMT amalgamated with captured image data. The system can be based on intelligent agents, where every agent explores the interaction between different medical data obtained by smart sensor devices using reinforcement learning as well as targets to detect diseases. The agents then collaborate to make a reliable conclusion about the detected diseases. Intensive experiments were conducted using medical data. The results show the importance of using intelligent agents for disease detection in healthcare decision-making. Moreover, collaboration increases the detection rate, with numerical results showing the superiority of the proposed framework compared with baseline solutions for disease detection.**



***Index Terms*—Communicable disease, correlation, multiagent system (MAS), smart sensor data.**

I. Introduction

SMART healthcare is a framework that leverages wearable devices, the Internet of Medical Things (IoMT), powerful machine learning algorithms, and wireless communication technology to connect people, resources, and organizations as well as then intelligently manage and respond to healthcare needs [1], [2]. Medical sensors, often referred to as IoMT, are a critical component of smart healthcare. IoMT may be able to be driving today's smart healthcare by leveraging cutting-edge technologies, such as artificial intelligence (AI), cloud computing, coupled with the emergent sixth generation (6G) mobile networks. The increasing use of IoMT devices, as well as data-driven apps, may be able to be contributing to positive effects ranging from improved user-health attitudes and early disease detection to higher-quality care and a more cost-effective smart healthcare ecosystem.

It may be expected that the integration of the Internet of the Things (IoT) with medical devices in a smart healthcare system will increase the quality and the efficiency of services for patients, especially for patients with chronic diseases who need continuous care. With the support of Internet communications, IoMT enables continuous monitoring of important physiological functions in, otherwise, healthy individuals, so that diseases may be able to be detected, and appropriate action may be able to be taken immediately. This may be able to be particularly important during pandemics, such as the recent Coronavirus-disease pandemic that raged globally [3], [4], where taking into account our advanced and technologically advanced healthcare systems, which tend to include both medical personnel with support systems, saw themselves under an absurd amount of stress [5]. The demand for a remote, autonomous, and ubiquitous IoMT architecture may be able to be greater than ever [6].

Manuscript received 21 April 2022; revised 9 July 2022; accepted 13 August 2022. Date of publication 2 September 2022; date of current version 12 January 2023. The associate editor coordinating the review of this article and approving it for publication was Dr. Hari P. Gupta. *(Corresponding author: Jerry Chun-Wei Lin.)*

Youcef Djenouri is with SINTEF Digital, 0610 Oslo, Norway (e-mail: youcef.djenouri@sintef.no).

Asma Belhadi is with the School of Economics, Innovation and Technology, Kristiania University College, 0107 Oslo, Norway (e-mail: asma.belhadi@kristiania.no).

Anis Yazidi is with the Department of Computer Science, OsloMet-Oslo Metropolitan University, 0167 Oslo, Norway, also with the Department of Neurosurgery, Oslo University Hospital, 0450 Oslo, Norway, and also with the Department of Computer Science, Norwegian University of Science and Technology, 7491 Trondheim, Norway (e-mail: anisy@oslomet.no).

Gautam Srivastava is with the Department of Math and Computer Science, Brandon University, Brandon, MB R7A 6A9, Canada, also with Research Centre for Interneural Computing, China Medical University, Taichung 404, Taiwan, and also with the Department of Computer Science and Math, Lebanese American University, Beirut 1102, Lebanon (e-mail: SRIVASTAVAG@brandonu.ca).

Pushpita Chatterjee is with the Department of Computer Science, Tennessee State University, Nashville, TN 37209 USA (e-mail: pushpita.c@ieee.org).

Jerry Chun-Wei Lin is with the Department of Computer Science, Electrical Engineering and Mathematical Sciences, Western Norway University of Applied Sciences, 5063 Bergen, Norway (e-mail: jerrylin@ieee.org).

Digital Object Identifier 10.1109/JSEN.2022.3202437

The integration of smart sensors and controls within the Internet also has turned any and all "so-called" cyber-physical systems (CPSs) into IoMT, which has recently emerged as one of the main driving forces behind the fourth industrial revolution nicknamed Industry 4.0 [7]. In these directions, healthcare itself may be able to be also undergoing a digital revolution through the integration of smart devices. On the other hand, the number of IoMT devices is expected to increase significantly over the next few years. Furthermore, the heterogeneity of the various IoMT components (network interfaces, data format, data semantics, and communication protocols) will lead to difficulties in interoperability and data protection [7]. In this regard, a ubiquitous, and collaborative health platform for all smart devices must be adaptable enough to accommodate all these concepts.

A. Motivation

Technologies based on AI are very promising in this regard and in medical applications in general [8], [9]. These include techniques, such as multiagent systems (MASs), networks that incorporate deep learning (DL), and advanced computational techniques, such as evolutionary computation (EC) or other well-known ones. DL is a well-known branch of AI that can involve the creation of complicated but complete models with a high number of layers and a large number of hyperparameters. These models are not only capable of learning from large amounts of data, but they can also directly extract important aspects from these huge amounts of data. Medical data analysis, especially disease detection, is a fascinating area of DL [10], [11], [12], [13]. For example, Coronavirus-disease pandemic samples were used to build an intelligent model to calculate infection rates [10]. The latter work uses both supervised and unsupervised learning methods, resulting in a 40% increase in detection speed. Transfer learning was used to evaluate pathogen frames and validate Coronavirus-disease pandemic instances with typical virus-based pneumonia [11]. The result highlights the value of using intelligent approaches for Coronavirus-disease pandemic diagnosis.

We may be able to also observe examples that have been substantially explored in the newer, fresh field of distributed DL [14], [15], [16], [17], [18] by studying various types of DL models that are well established in medicine as well as disease detection. The identification of illnesses is the primary objective of these technologies, particularly the distributed ones. This will assist medical professionals in making decisions that are acceptable and fair within the realm of medicine. The intricacy of the data is the single most critical barrier that makes disease detection more difficult than it would otherwise be. Indeed, diseases may be able to have different forms and manifestations that are difficult to detect. We are researching the possibility of developing a comprehensive framework that makes use of DL to get beyond these disadvantages and MASs.

The large number of hyperparameters provided by DL models is another significant obstacle to the disease detection process. The random selection of these values leads to a significant decrease in the overall performance during the learning phase. Moreover, the process of setting the parameters for such frameworks takes a long time, and there is no guarantee of satisfactory convergence. The effectiveness of EC in tackling complicated problems [19], [20] has led this research to tune the parameters of the proposed framework.

B. Contributions

To the best of our knowledge, this is the first paper to take an in-depth look at combining MASs, EC, and DL for disease detection. Below is a list of the major contributions.

1) We provide an collaborative system for disease detection (ALMOST), a new paradigm that uses DL, MAS, and EC to identify diseases. To learn from medical training data and different diseases, each and every agent uses its DL architecture. Each iteration of the architecture establishes communication between agents to share information and reduce the error learning rate.
2) We show how many convolutional neural networks (CNNs) may be able to work together to process large amounts of data in the medical domain. Several optimizations, such as batch normalization and dropout algorithms, ensure that the CNN processes medical data with great accuracy.
3) To intelligently explore the configuration space of different hyper parameter values, we propose a new evolutionary computational technique based on a genetic behavior. This approach improves the convergence of ALMOST in predicting diseases from medical data.
4) Extensive testing was conducted to demonstrate the applicability of ALMOST. The findings demonstrated that the ALMOST performs better than other well-known illness identification algorithms in terms of the quality of the information and also in terms of computation time when training large medical data.

From here on, this article is arranged as follows. Section II provides an in-depth examination of related studies on disease detection. Section III provides a comprehensive understanding of the ALMOST methodology. A performance evaluation of ALMOST is shown in Section IV. Section V discusses the major consequences of using ALMOST on medical data and the prospects for the future of the research. Section VI concludes this article.

II. Literature Review

A. AI-Based Solutions

Pattern mining [21], [22], [23] is one of the approaches to derive and reveal the potential relationships of the items in the databases. Nawaz *et al.* [24] investigated the use of pattern mining in the analysis of medical diseases. The set of Coronavirus-disease pandemic patient data is converted into a set of transactions, where each and every patient is represented by a transaction, and each Coronavirus-disease pandemic-based information related to the patient is represented by an item. A pattern mining algorithm is then applied to the set of transactions to extract relevant patterns. The latter was used to identify diseases based on the correlation between

medical data features. Wang *et al.* [10] automated the process of image assessment by exploring the segmentation as well as classification of DL-based architectures. Thus, we may be able to achieve a reasonable estimate of the always illusory Coronavirus-disease pandemic infection rate. Wang *et al.* [11] found viral pneumonia from more than 1000 images of pathogens. The experiments showed a clear benefit of using intelligent methods for disease diagnosis. Yan *et al.* [25] used a topological approach to analyze nonlinear dynamics to examine gait complexity fluctuations in Parkinson's disease patients to detect the Freezing-of-Gait episode. The 3-D acceleration data from eight individuals with symptoms are used to extract relevant features from the acceleration signals using topological analysis of the reconstructed process. Jain *et al.* [26] demonstrated the performances of using Inception-V3 for Coronavirus-disease pandemic disease identification through data enrichment. Sedik *et al.* [27] demonstrated the efficiency of using CNN in Coronavirus-disease pandemic identification. This article also demonstrated the importance of multimodal data, in which the authors collected medical data from multiple sources, including tomographic and X-ray images. Manoj *et al.* [28] took a look at an incentive-based system, where the Coronavirus pandemic could be better planned for using an incentive system using block chain. In this solution, governments globally can prepare and plan better strategies for fighting the virus using the availability of data from both a geographic point of view but also qualitative data around the disease. By being able to have more accurate and better data around the pandemic, and have it available in a decentralized fashion being managed by block chain, will allow a faster response time to virus outbreaks in the future.

B. Hybrid-Based Solutions

Chae *et al.* [29] predicted infectious diseases by successfully exploring long-term short-term memory with autoregressive moving average. The proposed model is improved by the ensemble learning mechanism. Therefore, more sources of information were collected and extracted from social networks. Ahuja *et al.* [30] implemented four DL architectures (ResNet18, ResNet50, ResNet101, and SqueezeNet) to capture Coronavirus-disease pandemic from medical lung CT-scan data. The models are pretrained using a large collection of images from different domains. The transfer learning mechanism is used to learn the Coronavirus-disease pandemic cases from the medical data. Wong *et al.* [31] analyzed the effect of data-driven solutions for infectious diseases. They explored the combination of various data management as well as AI techniques to help healthcare professionals mitigate the risk of disease detection and enable better diagnosis in a smart healthcare environment. Hirano *et al.* [32] classified the various diseases using the model for use in DL. The classification models developed are based on three types of medical images: photographic images, X-ray chest images, and retinopathy images. Three applications are then investigated, including skin cancer, transmissible diabetics, and pneumonia. Transfer learning with the adversarial neural network has been implemented. The transfer learning mechanism allows the model developed from various medical sources to be trained, and the adversarial network, which may be able to handle both non-targeted and targeted attacks and identify fake medical images. Jamshidi *et al.* [33] process multiple sources of medical data by exploring generative adversarial networks, extreme learning, and long-term short-term memory. This combination not only enables the handling of heterogeneous medical data but also increases the disease detection rate. Singh *et al.* [34] worked on developing a hybrid model based on both decompositions and DL for disease detection. Segments are created by applying the k-means algorithm to medical data. These segments are then fed into the CNN to predict diseases based on the original medical images. Shalbaf *et al.* [35] implemented 15 pre-trained DL models to automatically identify the Coronavirus-disease pandemic. These models are based on three well-known classification-based architectures, including Inception, ResNet, and DenseNet. Ensemble learning is then explored to merge the results obtained from these models using the majority voting strategy.

III. ALMOST: An Collaborative System for Disease Detection

A. Principle

We begin by explaining the most important aspects of ALMOST. ALMOST is a combination of many intelligent strategies for solving disease detection problems. The CNN is used for disease diagnosis. To correctly execute ALMOST in a distributed environment where each agent can benefit from the environment through the reinforcement learning paradigm, the use of an MAS is investigated. Since DL requires setting a large number of parameters, up to a million for some architectures, EC is used to determine the best settings for real-time processing. The components of ALMOST are discussed in the next parts.

B. Learning Phase

The learning phase is done with the help of CNNs [36]. CNN is a common type of deep architecture in computer vision applications, such as object detection and identification. In recent years, the adaptability of this method has helped both time series and text data. CNN is based on the concept of extracting features from matrix data using convolutional filters. Convolutional filters create a new image by applying a series of weights to the matrix data of each pixel in the image. As a result, a new image is generated. In addition, well-known operators for DL models, such as batch normalization and dropout, are used during training to increase the accuracy of the proposed framework. This was achieved using the dropout method. Batch normalization is one of the factors that help the network to converge faster, while the dropout method acts as a regulator that prevents the network from overfitting. These two methods are necessary for the network to achieve high accuracy. Below is a complete description of these components.

1) *Batch Normalization:* To efficiently train a large number of layers, we used the batch normalization technique in all steps of the training phase. With only a few epochs,

the learning process may be able to converge better. Batch normalization is performed after each and every convolutional layer in the CNN.

2) *Dropout:* It is a technique that allows you to avoid over adaptation during training. In each and every phase, the outputs of the neurons in the hidden layers are randomly skipped. Propagating a deep network with a constrained number of weights may be thought of as a straightforward method for bringing about convergence of predictions during the inference phase.

C. Multiagents Systems

The MAS is used to learn the different diseases in the training phase. The agents collaborate using the reinforcement learning process. Consider the tuple $\langle \mathcal{A}, \mathcal{S}, \mathcal{U}, \mathcal{R} \rangle$, an MAS is defined by $\mathcal{A}$. There are $\mathcal{A}$agents in total, and for the purposes of this discussion, each of them is treated as if it were an independent Markov decision process. A finite collection of environmental states is represented by the variable $\mathcal{S}$, a set of actions by the variable $\mathcal{U}$, and a reward function by the variable $\mathcal{R}$. The methods presented in $\mathcal{A}$ describe not only how each agent should behave given the current situation, but also how it should decide on the appropriate actions. For instance, in the process of disease detection, the mission of each agent is to identify the most effective tactic that maximizes the value of the target objective function, which may be the number of diseases that are accurately identified. We will discuss the many parts that make up the MAS that we have designed by the sections as follows.

1) *Environment:* The environment may be thought of as a collection of databases that hold a massive quantity of data that were gathered by various intelligent sensor devices. As a result, the environment is able to create certain conditions for training the agents and determine the optimal actions.
2) *State:* The next action of each and every agent is determined by the decisions made in earlier phases. The state of each agent is, thus, composed of two distinct parts: the data being processed and a collection of actions completed in the past. When calculating the size of the state space $\mathcal{S}$, the number of observations contained in the database is taken into account.
3) *Action:* It is the assignment of each and every observation in the database's decision-making behavior. For instance, a detection task is the assignment of each and every disease category.
4) *Reward:* It is essential to decide on an acceptable reward function. It makes it possible for each agent in $\mathcal{A}$ to have a more fruitful learning experience. Using data that included ground truths, we crafted a reward in response to the behaviors of the agent.

Therefore, the first thing each agent does is to perform a scan of the data collected by the ith intelligent sensor. This is denoted by the notation $\mathcal{A}_i$. It then computes the first observation of the ith intelligent sensor, and all future observations of that sensor. The ground truth for the first observation is used in the creation of a reward function specific to that choice. This approach is repeated for each of the ith smart sensor observations. As a result, a set of local options, denoted LD_i, is generated for each agent $\mathcal{A}_i$. The agents then learn from the local decisions $\{LD_i\}$ to optimally find the global decision. This learning is realized through the process of reinforcement learning, where the best agents that have a high score for their local decisions receive a reward.

D. Hyperparameters Optimization

To achieve optimal performance, we apply an evolutionary-based technique to optimize hyperparameters. The adaptation of the genetic algorithm is proposed due to its known balance of intensification and diversification. A complete description of the proposed algorithm is given for solving our hyper parameter optimization problem.

Let $\mathcal{HP} = \{\mathcal{HP}_1, \mathcal{HP}_2, \ldots, \mathcal{HP}_r\}$ be the set of hyperparameters, where r is the number of hyperparameters in the evolved ALMOST. Each $\mathcal{HP}_i$ represents a set of the possible values of the hyperparameter in question. The configuration space $\mathcal{C}$ is then defined by the set of all possible configurations, where each and every configuration is a vector. The possible values of all hyperparameters belong to $\mathcal{HP}$. When it comes to hyperparameter optimization, our methodology focuses on determining the ideal configuration that provides the highest level of accuracy. The configuration space is defined by the total number of possible hyper parameter values, as given in

$$|\mathcal{C}| = \prod_{i=1}^{r} |\mathcal{HP}_i|. \tag{1}$$

Because of the vastness of the configuration space, finding the ideal solutions requires considerable computational effort. Consider the epoch parameter, which has a max value of 1000, the error rate, which has a max value of 100, as well as the number of agents in the evolved model, which has a max value of 100; then, the search space will include ten million configurations, so it is not possible to apply exhaustive search methods in this case. To solve this challenge, evolutionary computational methods are used. Below are the main components of our solution.

1) Population Initialization: We are trying to distribute $|\mathcal{P}|$, which is the initial population, noted $\mathcal{P}$. This initial population should be uniformly distributed in the configuration space $\mathcal{C}$. Proper examination of each and every of the numerous alternative configurations that tend to cover most locations within $\mathcal{C}$ may be able to then be performed using this uniform distribution technique. We must first create the population, taking diversity into account.

This process begins with the random generation of an individual represented by a single $\mathcal{C}$ configuration. It is possible that after starting with this individual, we can generate more $|\mathcal{P}| - 1$, assuming that each newly formed individual is different from the previously generated individuals. We could use a distance measure between two successive configurations to evaluate dissimilarity based on the individuals formed in those configurations. This would be done based on the individuals generated in those configurations. The original population, denoted by the variable $\mathcal{P}$, should, in turn, be able to maximize

the diversification function shown in

$$\text{Diversify}(\mathcal{P}) = \sum_{i=1}^{|\mathcal{P}|} \sum_{j=1}^{|\mathcal{P}|} \text{Distance}(\mathcal{C}_i, \mathcal{C}_j) \tag{2}$$

where $\text{Distance}(\mathcal{C}_i, \mathcal{C}_j)$ is the distance between the configurations of the ith and the jth individuals, respectively.

2) Crossover: To produce new offspring, each and every of the two individuals in the current population goes through the following steps.

1) We start at 1 and work our way up to r, creating a random sequence of crossing points. At each of these points, we divide the intersection into its *left* and *right* halves, respectively.
2) Both the left-hand side of the original, which is copied to the left-hand side of the first descendant, and the right-hand side of the original, which is copied to the right-hand side of the second descendant, are descendants of each other.
3) The right-hand side of the second person is inherited by the first generation, while the left-hand side of the second individual is inherited by the second generation.

3) Mutation: Mutation facilitates the search for diversity. We use a strategy in which the value of a single parameter is randomly varied in every single one of the configurations currently in use. The mutation point is chosen randomly and can have a value between 1 and r, depending on the method. At each iteration, the value of the mutation point is changed in the descendants generated by the crossover operator.

4) Fitness Function: Identifying diseases with the highest possible degree of precision is the goal of the ALMOST framework. Therefore, to assign points to individuals within populations, we use the following function:

$$\text{Fitness}(\mathcal{C}_i) = \text{Detection}_{\text{ALMOST}}(\mathcal{C}_i). \tag{3}$$

Note that the following hold.

1) $\mathcal{C}_i$ is a representation of the configuration of the ith individual in the population.
2) $\text{Detection}_{\text{ALMOST}}(\mathcal{C}_i)$ indicates the ratio for disease detection using the developed ALMOST based on the $\mathcal{C}_i$.

Based on these actions, we presented the following method for optimizing hyperparameter values. In the beginning, the initial population size, defined as $|\mathcal{P}|$, is randomly generated. Then, each and every individual is constructed using population initialization. Then, mutation, and crossover with mutation and crossover rates (Mr and Cr), is used to generate configurations from $\mathcal{C}$. To keep the population size constant, each individual is evaluated against the fitness function, focusing on maintaining the highest quality $|\mathcal{P}|$ individuals. All others are now deleted. This procedure is then continued endlessly until the max number of iterations (IMAX) is reached.

E. Description

Algorithm 1 shows the pseudocode of ALMOST for each and every agent. The process begins by building the model for use in DL represented by the CNN with batch normalization and dropout layers (from lines 4 to 5). The batches of data are created from the input images I in line 6. Then, the genetic algorithm is applied to optimize the hyperparameters of the model for use in DL by performing the training phase on the created batches (line 7). The inference phase is then run on the trained model to identify the disease of the new image (line 8). The output of the algorithm is the set of detected diseases DD and the trained disease detection model (line 11).

Algorithm 1 ALMOST Pseudocode for Each Agent

1: **Input**: $\mathcal{I} = \{\mathcal{I}_1, \mathcal{I}_2, \ldots, \mathcal{I}_m\}$: the set of m images collected from sensors.
2: **Output**: $model$: The trained model to detect the disease; DD: the set of the disease detected from I.
3: $model \leftarrow CNN()$;
4: $model \leftarrow model \cup BatchNormalization()$;
5: $model \leftarrow model \cup Dropout()$
6: $Batches \leftarrow CreatingBatches(D)$;
7: $Hyper\_Param \leftarrow GA(fit(model, Batches))$;
8: $DD \leftarrow Inference(I_{new}, model, Hyper\_Param)$;
9: **return** $< model, DD >$.

IV. Performance Evaluation

Extensive testing was performed on known medical datasets specifically designed for disease detection applications to validate the use of the proposed ALMOST framework. Experiments were conducted using a desktop computer equipped with 16 GB of primary memory and an Intel Core i7 processor for optimal performance. PythonTorch was used for the actual implementation of each algorithm. We used the Kvasir [37] medical database to validate the applicability of ALMOST in disease detection, for disease data for the human digestive system. The aim is to automate the detection of endoscopic findings in the esophagus, stomach, intestines, and rectum. It is available in two versions. The first version, called Kvasir (v1), consists of 4000 images grouped into eight classes showing anatomical landmarks, pathological findings, or endoscopic procedures. The second version, called Kvasir (v2), expands on the first version and consists of 8000 images with the same number of classes.

A. Parameter Setting

In ALMOST, several parameters need to be optimized, including the number of agents, the number of generations, the crossover and mutation rates, and the population size. Choosing optimal values for these parameters is critical for better performance of the ALMOST framework. In this experiment, we analyze the behavior of ALMOST at different values for the number of agents, number of generations, crossover rate, and the mutation rate. We varied the number of agents from 2 to 20, the number of generations and population size from 10 to 100, and the crossover rate and mutation rate from 0.01 to 0.99. The behavior of ALMOST may be able to be summarized as follows.

1) *Number of Agents:* The experiments showed that when we vary the number of agents from 2 to 20, the accuracy

TABLE I
SUMMARY OF PARAMETER SETTING OF ALMOST

Dataset	$\|A\|$	**IMAX**	$\|P\|$	Cr	Mr
Kvasir (v1)	5	45	85	0.35	0.53
Kvasir (v2)	8	58	93	0.47	0.61

TABLE II
ALMOST VERSUS DISEASE DETECTION SOLUTIONS

$Dataset\_\|Images\|$	**ALMOST**		**InceptionResNet**		**DenseNet**	
	F1	Accuracy	F1	Accuracy	F1	Accuracy
Kvasir(V1)_1000	0.53	0.57	0.48	0.51	0.47	0.49
Kvasir(V1)_2000	0.56	0.59	0.50	0.53	0.50	0.51
Kvasir(V1)_3000	0.58	0.63	0.52	0.55	0.52	0.53
Kvasir(V1)_4000	0.63	0.66	0.55	0.58	0.54	0.54
Kvasir(V2)_1000	0.57	0.62	0.56	0.56	0.53	0.54
Kvasir(V2)_2000	0.64	0.66	0.59	0.60	0.54	0.57
Kvasir(V2)_3000	0.69	0.73	0.60	0.60	0.58	0.61
Kvasir(V2)_4000	0.75	0.77	0.65	0.69	0.63	0.64
Kvasir(V2)_5000	0.80	0.84	0.68	0.72	0.65	0.67
Kvasir(V2)_6000	0.83	0.86	0.72	0.74	0.66	0.69
Kvasir(V2)_7000	0.87	0.91	0.75	0.77	0.71	0.72
Kvasir(V2)_8000	0.92	0.96	0.77	0.79	0.72	0.75

of ALMOST increases until five agents for Kvasir (V1) and eight agents for Kvasir (V2) where stabilization of the accuracy is observed.

2) *Number Generations:* The experiments showed that when we vary the number of generations from 10 to 100, the accuracy of ALMOST increases until 45 generations for Kvasir (V1), and 58 generations for Kvasir (V2) where stabilization of the accuracy is observed.
3) *Population Size:* The experiments showed that when we vary the population size from 10 to 100, the accuracy of ALMOST increases until 85 individuals for Kvasir (V1), and 93 individuals for Kvasir (V2) where stabilization of the accuracy is observed.
4) *Crossover Rate:* The experiments showed that when we vary the crossover from 0.01 to 0.99, the accuracy of ALMOST increases until 0.35 for Kvasir (V1), as well as 0.47 for Kvasir (V2) where stabilization of the accuracy is observed.
5) *Mutation Rate:* The experiments showed when we vary the mutation from 0.01 to 0.99, the accuracy of ALMOST increases until 0.53 for Kvasir (V1), as well as 0.61 for Kvasir (V2) where the stabilization of the accuracy is observed.

Table I gives the optimal values of the parameters used in ALMOST for both Kvasir (v1) and Kvasir (v2). The next experiments aim to validate the usefulness of the proposed ALMOST framework for disease detection. To reach this conclusion, an intensive analysis was performed by comparing ALMOST with the baseline solutions InceptionResNet [34] and DenseNet [35]). The detailed results with a full explanation are shown below.

B. Quality of Outputs

Table II shows the quality of results from ALMOST and the baseline solutions: InceptionResNet, DenseNet on Kvasir (V1), and Kvasir (V2). We varied the percentage of images used for training from 1000 to 4000 for Kvasir (V1) and from 1000 to 8000 images for Kvasir (V2). We then calculate

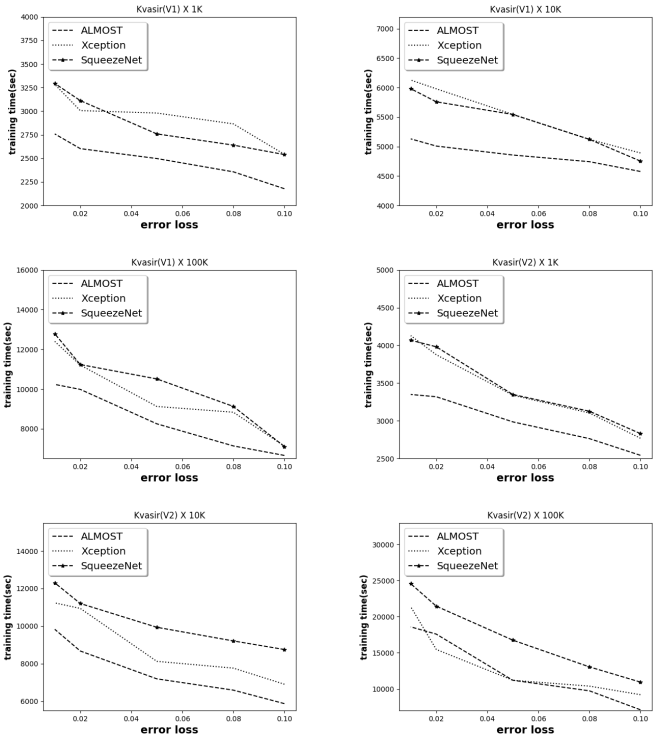


Fig. 1. ALMOST versus advanced disease detection solutions with different numbers of error loss values (0.10, 0.08, 0.05, 0.02, and 0.01).

the quality of the results represented by the F1 and accuracy formulas. The results show the superiority of ALMOST compared with the baseline solutions for all scenarios. Thus, the accuracy of ALMOST is 0.96 when all the data from Kvasir (V2) are processed, while the accuracy of the two solutions is less than 0.80 when the same data are trained. This great performance is due to the efficient components of ALMOST represented by the DL solution and the MASs, and the accurate way of the hyperoptimization process. Fig. 2 shows a case study of ALMOST. The first three images are considered as esophagitis disease, and the second three images are considered as polyps disease, where the last three images are considered as ulcerative colitis disease.

C. ALMOST for Large-Scale Data

In the next experiment, we will examine the scalability of ALMOST compared with the baseline solutions when it comes to processing large amounts of data. For comparison, we will use Xception [38] and SqueezeNet [39]. These algorithms have proven their usefulness in a variety of different contexts, including training huge datasets. Different training scenarios with different data sizes of Kvasir (v1) and Kvasir (v2) are run. Data duplication is generated by multiplying Kvasir (v1) and Kvasir (v2) multiple times (1000, 10 000, and 100 000). For each and every redundant sample, changes are generated using a generative adversarial network. We varied the error loss to be optimized from 0.10 to 0.01, and the results are given in Fig. 1. From these results, we may be able to see the clear superiority of ALMOST over the other two solutions in terms of training time. This performance may be able to be explained by the fact that ALMOST is optimized DL where

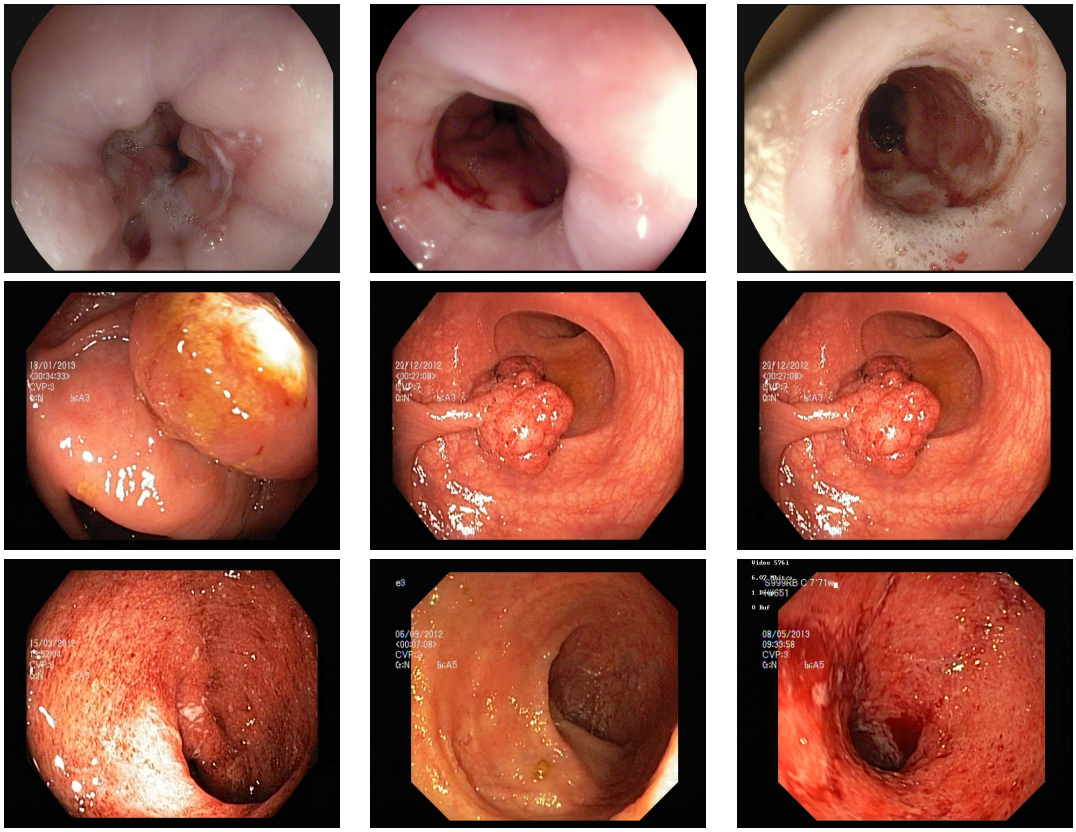


Fig. 2. Case study of ALMOST: the first three images are considered as esophagitis disease, and the second three images are considered as polyps disease, where the last three images are considered as ulcerative colitis disease.

the collaboration between the different agents accelerates the training process.

V. Discussion and Future Direction

The primary benefits of applying ALMOST to disease detection data are presented in this section. We also make some recommendations for how to improve the ALMOST framework.

1) The effective combination of intelligent technologies in the form of DL, MASs, and metaheuristics leads to a high level of precision. For real-time medical data management and disease detection, runtime performance is still a challenge. The development of hybrid systems that combine evolutionary and exact approaches [40] to improve the performance of ALMOST could be an interesting avenue.
2) The proposed methodology provides better results than previous approaches. It would extremely fascinating to study the results of ALMOST for other smart healthcare applications, such as brain tumor detection [41], as well as surgery [42].
3) Output interpretation is a challenge in ALMOST. It relies on black-box models that do not implicitly explain the process of output interpretation. Healthcare practitioners need to understand how the given output is produced to trust it. This problem is being addressed by the emerging discipline of explainable AI (XAI). We intend to incorporate XAI approaches into ALMOST. This will allow for a more accurate interpretation of the results from ALMOST.

VI. Conclusion

In this article, an intelligent collaborative system for disease detection is proposed. It studied the different interactions between the medical data using intelligent agents with an efficient reinforcement learning mechanism. This enables significant determination of various diseases in healthcare systems. The proposed system was tested on different medical datasets. The initial results showed the usefulness of using intelligent agents for healthcare diagnosis. The numerical results can be seen to also clearly visualize the strength of our proposed framework when directly compared with baseline methodologies when focusing on the rate of disease detection.

References

[1] R. Yadav *et al.*, "Smart healthcare: RL-based task offloading scheme for edge-enable sensor networks," *IEEE Sensors J.*, vol. 21, no. 22, pp. 24910–24918, Nov. 2021.
[2] L. Babangida, T. Perumal, N. Mustapha, and R. Yaakob, "Internet of Things (IoT) based activity recognition strategies in smart homes: A review," *IEEE Sensors J.*, vol. 22, no. 9, pp. 8327–8336, May 2022.
[3] Y. Han and H. Yang, "The transmission and diagnosis of 2019 novel coronavirus infection disease (COVID-19): A Chinese perspective," *J. Med. Virol.*, vol. 92, no. 6, pp. 639–644, 2020.
[4] J. He, Y. Guo, R. Mao, and J. Zhang, "Proportion of asymptomatic coronavirus disease 2019: A systematic review and meta-analysis," *J. Med. Virol.*, vol. 93, no. 2, pp. 820–830, 2021.
[5] E. J. Emanuel *et al.*, "Fair allocation of scarce medical resources in the time of COVID-19," *New England J. Med.*, vol. 382, no. 21, pp. 2049–2055, May 2020.

[6] T. Yang, M. Gentile, C.-F. Shen, and C.-M. Cheng, "Combining point-of-care diagnostics and internet of medical things (IoMT) to combat the COVID-19 pandemic," *Diagnostics*, vol. 10, no. 4, p. 224, Apr. 2020.

[7] L. M. Camarinha-Matos, R. Fornasiero, and H. Afsarmanesh, "Collaborative networks as a core enabler of industry 4.0," in *Collaboration in a Data-Rich World (PRO-VE)* (IFIP Advances in Information and Communication Technology), vol. 506, L. Camarinha-Matos, H. Afsarmanesh, and R. Fornasiero, Eds. Cham, Switzerland: Springer, 2017, doi: 10.1007/978-3-319-65151-4_1.

[8] S. Pouyanfar *et al.*, "A survey on deep learning: Algorithms, techniques, and applications," *ACM Comput. Surv.*, vol. 51, no. 5, pp. 1–36, Sep. 2018.

[9] A. Sharma *et al.*, "Multi-agent system applications to fight COVID-19 pandemic," *Apollo Med.*, vol. 17, no. 5, p. 41, 2020.

[10] B. Wang *et al.*, "AI-assisted CT imaging analysis for COVID-19 screening: Building and deploying a medical AI system," *Appl. Soft Comput.*, vol. 98, Jan. 2021, Art. no. 106897.

[11] S. Wang *et al.*, "A deep learning algorithm using CT images to screen for corona virus disease (COVID-19)," *Eur. Radiol.*, vol. 31, pp. 6096–6104, Feb. 2021.

[12] D. M. Khan, K. Masroor, M. F. M. Jailani, N. Yahya, M. Z. Yusoff, and S. M. Khan, "Development of wavelet coherence EEG as a biomarker for diagnosis of major depressive disorder," *IEEE Sensors J.*, vol. 22, no. 5, pp. 4315–4325, Mar. 2022.

[13] R. Wang *et al.*, "A standalone and portable microfluidic imaging detection system with embedded computing for point-of-care diagnostics," *IEEE Sensors J.*, vol. 22, no. 6, pp. 6116–6123, Mar. 2022.

[14] N. Balachandar, K. Chang, J. Kalpathy-Cramer, and D. L. Rubin, "Accounting for data variability in multi-institutional distributed deep learning for medical imaging," *J. Amer. Med. Inform. Assoc.*, vol. 27, no. 5, pp. 700–708, May 2020.

[15] S. S. Roy, K. Samanta, S. Modak, S. Chatterjee, and R. Bose, "Cross spectrum aided deep feature extraction based neuromuscular disease detection framework," *IEEE Sensors Lett.*, vol. 4, no. 6, pp. 1–4, Jun. 2020.

[16] H. Ku, W. Susilo, Y. Zhang, W. Liu, and M. Zhang, "Privacy-preserving federated learning in medical diagnosis with homomorphic re-encryption," *Comput. Standards Interfaces*, vol. 80, Mar. 2022, Art. no. 103583.

[17] X. Xu, H. Tian, X. Zhang, L. Qi, Q. He, and W. Dou, "DisCOV: Distributed COVID-19 detection on X-ray images with edge-cloud collaboration," *IEEE Trans. Services Comput.*, vol. 15, no. 3, pp. 1206–1219, May 2022.

[18] R. Dwivedi, S. Dey, C. Chakraborty, and S. Tiwari, "Grape disease detection network based on multi-task learning and attention features," *IEEE Sensors J.*, vol. 21, no. 16, pp. 17573–17580, Aug. 2021.

[19] U. Ahmed, J. C.-W. Lin, G. Srivastava, R. Yasin, and Y. Djenouri, "An evolutionary model to mine high expected utility patterns from uncertain databases," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 5, no. 1, pp. 19–28, Feb. 2021.

[20] P. Srinivas and R. Katarya, "HyOPTXg: OPTUNA hyper-parameter optimization framework for predicting cardiovascular disease using XGBoost," *Biomed. Signal Process. Control*, vol. 73, Mar. 2022, Art. no. 103456.

[21] C.-W. Lin, T.-P. Hong, G.-C. Lan, J.-W. Han, and W.-Y. Lin, "Incrementally mining high utility patterns based on pre-large concept," *Appl. Intell.*, vol. 40, no. 2, pp. 343–357, Mar. 2014.

[22] J. C.-W. Lin, W. Gan, P. Fournier-Viger, T.-P. Hong, and J. Zhan, "Efficient mining of high-utility itemsets using multiple minimum utility thresholds," *Knowl.-Based Syst.*, vol. 113, pp. 100–115, Dec. 2016.

[23] W. Gan, J. C.-W. Lin, J. Zhang, P. Fournier-Viger, H.-C. Chao, and P. S. Yu, "Fast utility mining on sequence data," *IEEE Trans. Cybern.*, vol. 51, no. 2, pp. 487–500, Feb. 2021.

[24] M. S. Nawaz, P. Fournier-Viger, A. Shojaee, and H. Fujita, "Using artificial intelligence techniques for COVID-19 genome analysis," *Int. J. Speech Technol.*, vol. 51, no. 5, pp. 3086–3103, May 2021.

[25] Y. Yan *et al.*, "Topological descriptors of gait nonlinear dynamics toward freezing-of-gait episodes recognition in Parkinson's disease," *IEEE Sensors J.*, vol. 22, no. 5, pp. 4294–4304, Mar. 2022.

[26] R. Jain, M. Gupta, S. Taneja, and D. J. Hemanth, "Deep learning based detection and analysis of COVID-19 on chest X-ray images," *Appl. Intell.*, vol. 51, pp. 1690–1700, Oct. 22021.

[27] A. Sedik, M. Hammad, F. E. Abd El-Samie, B. B. Gupta, and A. A. Abd El-Latif, "Efficient deep learning approach for augmented detection of Coronavirus disease," *Neural Comput. Appl.*, vol. 34, pp. 11423–11440, Jan. 2021.

[28] M. Manoj, G. Srivastava, S. R. K. Somayaji, T. R. Gadekallu, P. K. R. Maddikunta, and S. Bhattacharya, "An incentive based approach for COVID-19 planning using blockchain technology," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, Dec. 2020, pp. 1–6.

[29] S. Chae, S. Kwon, and D. Lee, "Predicting infectious disease using deep learning and big data," *Int. J. Environ. Res. Public Health*, vol. 15, no. 8, p. 1596, Jul. 2018.

[30] S. Ahuja, B. K. Panigrahi, N. Dey, V. Rajinikanth, and T. K. Gandhi, "Deep transfer learning-based automated detection of COVID-19 from lung CT scan slices," *Appl. Intell.*, vol. 51, no. 1, pp. 571–585, 2021.

[31] Z. S. Y. Wong, J. Zhou, and Q. Zhang, "Artificial intelligence for infectious disease big data analytics," *Infection, Disease Health*, vol. 24, no. 1, pp. 44–48, Feb. 2019.

[32] H. Hirano, A. Minagi, and K. Takemoto, "Universal adversarial attacks on deep neural networks for medical image classification," *BMC Med. Imag.*, vol. 21, no. 1, pp. 1–13, Dec. 2021.

[33] M. Jamshidi *et al.*, "Artificial intelligence and COVID-19: Deep learning approaches for diagnosis and treatment," *IEEE Access*, vol. 8, pp. 109581–109595, 2020.

[34] P. Singh, A. Verma, and J. S. R. Alex, "Disease and pest infection detection in coconut tree through deep learning techniques," *Comput. Electron. Agricult.*, vol. 182, Mar. 2021, Art. no. 105986.

[35] P. Gifani, A. Shalbaf, and M. Vafaeezadeh, "Automated detection of COVID-19 using ensemble of transfer learning with deep convolutional neural network based on CT scans," *Int. J. Comput. Assist. Radiol. Surgery*, vol. 16, no. 1, pp. 115–123, Jan. 2021.

[36] D. Moolchandani, A. Kumar, and S. R. Sarangi, "Accelerating CNN inference on ASICs: A survey," *J. Syst. Archit.*, vol. 113, Feb. 2021, Art. no. 101887.

[37] K. Pogorelov *et al.*, "Kvasir: A multi-class image dataset for computer aided gastrointestinal disease detection," in *Proc. 8th ACM Multimedia Syst. Conf.*, 2017, pp. 164–169.

[38] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1251–1258.

[39] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size," 2016, *arXiv:1602.07360*.

[40] Y. Djenouri and M. Comuzzi, "Combining Apriori heuristic and bio-inspired algorithms for solving the frequent itemsets mining problem," *Inf. Sci.*, vol. 420, pp. 1–15, Dec. 2017.

[41] M. Woźniak, J. Siłka, and M. Wieczorek, "Deep neural network correlation learning mechanism for CT brain tumor detection," *Neural Comput. Appl.*, vol. 1, pp. 1–16, Mar. 2021, doi: 10.1007/s00521-021-05841-x.

[42] P. N. Ramkumar *et al.*, "Clinical and research medical applications of artificial intelligence," *Arthroscopy: J. Arthroscopic Related Surg.*, vol. 37, no. 5, pp. 1694–1697, 2021.