OncoVision

Rocco Iuliano, Simone Delle Porta

ACM Reference Format:
Rocco Iuliano, Simone Delle Porta. 2023. OncoVision. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 PROJECT CONTEXT

Addressing health-related issues represents the next frontier of artificial intelligence. Regarding the healthcare context, several attempts have already been conducted to develop machine learning solutions, genetic algorithms and others. For instance, in the Aravind Eye Care System in India, ophthalmologists and computer scientists are working together to test and deploy an automated image classification system to screen millions of retinal photographs of diabetic patients. Since the mid-twentieth century, researchers have proposed and developed many clinical decision-support systems to help physicians. Rule-based approaches were proposed in the 1970s that allow us to:

(1) diagnose diseases;
(2) choose appropriate treatments;
(3) provide interpretations of clinical reasoning;
(4) assist physicians in generating diagnostic hypotheses in complex patient cases.

It seems a good approach but it presents several problems, such as:

(1) it is costly to build;
(2) it requires explicit expressions of decision rules and require human-authored updates;
(3) it is difficult to encode higher-order interactions among different pieces of knowledge authored by different experts;
(4) the performance of the systems is comprehensible only by a medical knowledge;
(5) it was difficult to implement a system that integrates deterministic and probabilistic reasoning to narrow down relevant clinical context, prioritize diagnostic hypotheses, and recommend therapy.

Instead, recent AI research has leveraged machine learning methods, which can account for complex interactions, to identify patterns from the data. Moreover, machine-learning methods enable the development of AI applications that facilitate the discovery of previously unrecognized patterns in the data without the necessity to specify decision rules for each specific task [6]. Therefore, the main goal of an AI system in Medicine is to help the physicians in the clinical process, namely detect something, cluster the patient on some criteria, etc. On the other hand, AI systems pose new risks for medicine. Failures in medical AI could erode public trust in Healthcare. For example, bias in AI can deliver erroneous medical evaluations. Moreover, AI models magnify existing cyber-security risks, potentially threatening patient privacy and confidentiality. In general, in a software project, we need three types of expert groups:

- the group developing the algorithm;
- a group of validators;
- the operational staff.

These groups are also needed in the healthcare sector to overcome the following 3 key challenges in AI:

- Conceptual challenges in formulating a problem that AI can solve;
- Technical challenges in implementing an AI solution;
- Humanistic challenges regarding AI's social and ethical implications.

Furthermore, the incapability to address these challenges could erode public trust in medical AI, which could in turn undermine trust in Healthcare institutions themselves. In the end, unlike physicians, AI cannot draw upon "common sense" or "clinical intuition" but using good data and the correct AI model configuration, we can obtain a good performance [4]. In this project, we focused on skin cancer, in particular, the melanoma. Melanoma is a malignancy of melanocytes, which are pigment-producing cells of neuroectodermal origin that can be found throughout the body (including in the skin, iris and rectum). The cutaneous form of the disease is common in the Western world causing the majority (75%) of deaths related to skin cancer; indeed, its global incidence is 15–25 per 100,000 individuals. Survival rates in patients with melanoma (cumulative for all forms) have shown huge differences between countries in Europe, ranging between <50% in Eastern Europe to >90% in northern and central Europe for 5-year survival after primary diagnosis [5]. Figure 1 shows the incidence and mortality of cutaneous melanoma in the world. In according with Istituto Superiore di Sanità, the general melanoma diagnosis process is the following: the patient will conduct a dermoscopic examination and if the doctor has some suspects, he takes a skin sample in order to conduct histological analysis and then formulate the diagnosis. The biggest problem with this cancer type is a lack of early detection that does not allow therapy to treat the disease on time. That is why we decided to develop a melanoma detection AI system to help physicians with this disease. We have compared the performance obtained by our model with the proposed one by Di Biasi et al [3].

1.1 Related project

Di Biasi et al. proposed a system that combines Genetic Algorithms (GAs) with Convolutional Neural Networks (CNNs) to detect melanoma. They used GA for improving the architecture of CNN. Indeed, they defined a population of neural networks (NN) that are codified in vectors where each vector element represents a type of layer or pre-processing routine. Whereas for the hyperparameters of each individual, they fixed them to these values: *'sgdm', 'MaxEpochs', 16, 'MiniBatchSize', 12, 'Shuffle', 'every-epoch',*

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

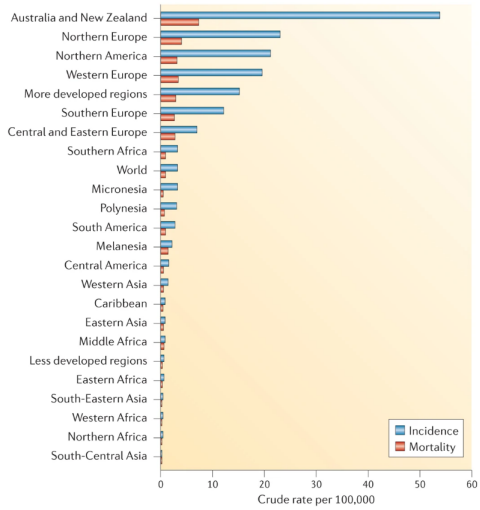

Figure 1: Incidence and mortality of cutaneous melanoma[5]

'InitialLe-arnRate', 0.0001, 'ExecutionEnvironment', 'auto'.
The authors defined the following constraints to the Genetic Algorithm in order to obtain a correct population of NN at each evolution step:

- The first gene of each entity must be an image input (II) or one of the pre-processing routines;
- If the gene q is a pre-processing routine, then the gene q + 1 must be a II layer or another pre-processing layer;
- The latest gene of an entity must be a classification layer.

Furthermore, they forced the possible values that the genes of an individual can assume in this range: *Convolution, ReLu, Cross Channel Normalization, Max Pooling Grouped Convolution, Fully Connected Layer, Dropout and Softmax.*
The authors stopped the GA after 100 evolutional steps or if no accuracy improvement was detected for ten consecutive evolution steps. In conclusion, their dataset contains skin images divided in two classes, melanoma (positive class) and moles (negative class). The number of instances for the positive class is 70 and 100 for the negative one.

2 PROJECT GOALS

In this work we will reply to the following research questions:

RQ$_1$: *What are the limitations of the selected approach?*

We decided to investigate this aspect to understand problems that afflict the selected approach to develop a better model. The results of this **RQ** influence the input of **RQ$_3$** because we are going to try to define a model that is less affected by these problems.

RQ$_2$: *What are the most useful data types and what are the data problems in the melanoma domain?*

We decided to investigate this aspect to understand which are the relevant data types to train the model for melanoma detection and which are the problems with the data. The results of this **RQ** influence **RQ$_4$** because, based on data problems, we can identify the techniques to improve their quality. Moreover, this **RQ** influences **RQ$_1$** because we can understand if the detected problems are also in the existing approach.

RQ$_3$: *How to improve melanoma detection model performance?*

We decided to investigate this aspect because, based on the other **RQs**' results, we can understand the changes to apply to the model to improve performance.

RQ$_4$: *How to raise the data quality?*

We decided to investigate this aspect to understand which are the techniques for improving data quality to highlight relevant features that the model will learn for making more precise predictions.
Based on the previous research questions, we have defined the following project goals:

(1) Conduct a detailed investigation of the baseline approach selected from the literature to understand the performance of the approach and its limitation;
(2) Understand the problems related to the datasets, namely lack of relevant features, few samples, low data quality, etc;
(3) Definition of an AI pipeline that might be used for cancer detection and is not affected by the problems which the baseline approach selected from the literature suffers.

Our project is available on GitHub at this link: https://github.com/Rocco000/OncoVision

3 AI MODEL SELECTION

In this study, we decided to use Convolutional Neural Network (CNN) because it represents various benefits. Before all, we must first explain its features and how it works. CNN is a classic artificial neural network (ANN) with input, hidden and output layers. The input layer propagates the input in the network, instead, the output layer provides the model prediction. Each layer can contain one or more neurons linked with some neurons of the previous layer. Moreover, each link has an associated weight. Each neuron calculates its activation as the weighted sum of its inputs and then adds a bias value. After which, it applies an activation function to the obtained value. CNN has an important peculiarity which is the convolutional layer. How affirmed by Kun-Hsing Yu et al [6], this layer type is useful for extracting spatial or temporal relations in an image and allows us to summarize and transform clusters of image pixels to extract high-level features. Moreover, according with Albawi et al [1], CNN has these benefits:

(1) The assumption about problems that are solved by CNN is that the problems have no spatially dependent features, therefore, it provides an opportunity to detect and recognize features regardless of their positions in the image. For example, in face detection, we do not need to pay attention to where the faces are located in the images;

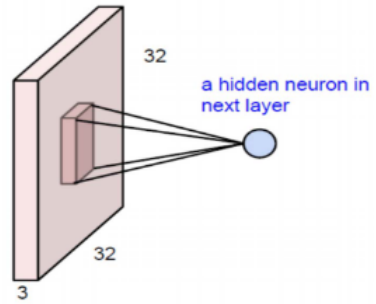

Figure 2: Example of how CNN hidden layer's neuron is responsible for a local region in the picture

(2) Each layer can detect a particular feature when input propagates toward the network. For example, the first layer detects the edges, the second detect shapes, etc;
(3) It can reduce the number of parameters to improve the computational complexity of the training process.

To understand the third point we can use an image detection problem as an example. Consider an image 32x32x3 as input. If we use classic ANN, for connecting the input layer with a hidden layer's neuron we need 32x32x3 weight connections and this number must be multiplied by the number of neurons in the hidden layer. This amount of parameters is required only to connect the input layer with the first hidden layer's neurons! Instead, in the convolutional neural network, each neuron in the hidden layer is responsible for an image local region as shown in Figure 2. It means that a neuron of the next layer can only get input from the corresponding part of the previous layer (i.e. the neuron is connected only with specific neurons of the previous layer). For example, it can only be connected to 5×5 neurons. Therefore, if we want 32×32 neurons in the next layer, we will have 5×5×3 by 32x32 connections, which is 76,800 connections compared with 3,145,728 in ANN. Another assumption for simplification; is to keep the local connection weights fixed for the entire neurons of the next layer. This will connect the neighbour neurons in the next layer with exactly the same weight to the local region of the previous layer. With this assumption, the layers are similar to a window of 5×5×3 sliding on input neurons and mapping the generated output to the corresponding place. For this reason, this type of layer is named convolutional and it looks like a filter in image processing. Therefore, each layer can be associated with different filters. The network architecture and the configuration of the training algorithm hyperparameters of our project have been defined by the genetic algorithm output.

3.1 PEAS Specification

The problem that we have treated has the PEAS specification that is shown in Table 1.

Performance	For evaluating our model, we used these metrics: accuracy, recall, precision and F1-score. In particular, we will focus on recall because we need to minimize the number of false negatives.
Environment	Our model operates in an environment that has these features: (1) **Fully observable**: because the sensors provide our model full environment state; (2) **Static**: because the environment doesn't change after an action that our model did or during the time; (3) **Discrete**: because the environment limits the perceptions and actions of our model; (4) **Episodic**: because an action that our model did, is independent of the previous actions; (5) **Single-agent**
Actuators	The model interacts with the environment through the standard output of the machine
Sensors	The model can receive the environment input through the standard input of the machine

Table 1: PEAS Specification

4 METHODICAL STEPS

Based on the goals set defined in Section 2, the methodical steps that we have conducted to address it are:

- Define a survey for physicians who are experts on cancer disease to understand data problems, data types that are useful for defining a good model and how to measure data quality. We conducted this step by using Google Forms;
- Re-implement the existing approach because its source code is not available. We need to do this in order to compare our model performance with the performance of the existing approach;
- Investigate some image processing techniques that are useful to improve data quality, like contrast and brightness improvement techniques. We chose other techniques based on survey results and all of these are implemented directly in the model;
- Search one or more new datasets in order to train and test our model and test the existing approach. We conducted these researches by using Kaggle;
- Develop three genetic algorithms (GAs) that produce a population of convolutional neural networks (CNNs) that are optimized from three points of view:
  - The goal of the first one is to improve the network hyperparameters;
  - The goal of the second one is to improve the network architecture. After that, we selected the best individual and applied the Grid Search algorithm on it in order to improve hyperparameters;
  - The goal of the last one is to improve both network hyperparameters and the architecture.

  At the end of each Genetic Algorithm, we will select the best individual in the last population obtained based on the evaluation metrics. For developing these alternatives we used Python and its libraries.

Finally, we compared the results obtained by the three different versions of our model and selected the best one. After that, we compared the selected one with the existing approach. Figure 3 summarizes the methodical steps that we conducted.

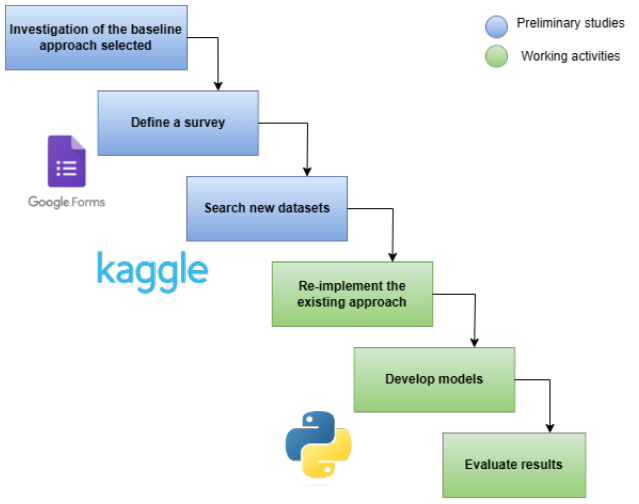

Figure 3: Methodology

4.1 Survey

The survey has two goals: the first is that it can guide us to select useful data for training the model and the techniques for improving image quality. The second one is to understand the problems that the melanoma diagnosis process suffers and, as a consequence, the problems that the selected approach suffers. The physicians are involved by sharing the survey's Google link to involve as many as possible doctors. The survey's questions are:

(1) Q_1: What is your role?
(2) Q_2: How many experience years do you have?
(3) Q_3: How much is incisive the dermoscopic image to a melanoma diagnosis? The possible response is a value between one and five where one represents "few relevant" and five represents "much relevant";
(4) Q_4: How much is incisive the histological image to a melanoma diagnosis? The possible response is a value between one and five where one represents "few relevant" and five represents "much relevant";
(5) Q_5: How reliable is the diagnosis if we have only the dermoscopic image? The possible response is a value between one and five where one represents "few reliable" and five represents "much reliable";
(6) Q_6: How reliable is the diagnosis if we have only the histological image? The possible response is a value between one and five where one represents "few reliable" and five represents "much reliable";
(7) Q_7: How reliable is the diagnosis if we have both dermoscopic and histological images? The possible response is a value between one and five where one represents "few reliable" and five represents "much reliable";
(8) Q_8: What are the problems that occur during the acquisition of dermoscopic images? The possible choices are: lack of light, presence of hair, movement of the patient, incorrect position of dermatoscope and other;
(9) Q_9: What are the problems that occur during the acquisition of histological images? The possible choices are: poor preparation of skin tissue, noise in the skin tissue, non-uniform sectioning and other;
(10) Q_{10}: Which are the useful metrics for evaluating a dermoscopic image quality? The possible choices are: Signal-to-noise ratio, Contrast-to-noise ratio, image sharpness and other;
(11) Q_{11}: What are the relevant dermoscopic image features on which you focus more? It can be replied through a text box;
(12) Q_{12}: What are the relevant histological image features on which you focus more? It can be replied through a text box;
(13) Q_{13}: Do you think there are other relevant problems in melanoma diagnosis that the survey did not treat? It can be replied through a text box.

The questions gave us the following informations:

- Q_1 and Q_2 allowed us to understand the role and experience years of the participants.
- Q_3 and Q_4 allowed us to understand the most relevant data type.
- Q_5, Q_6, and Q_7 allowed us to understand how much is reliable the prediction having a certain type of data.
- Q_8 and Q_9 allowed us to understand the problems that occur during data acquisition.
- Q_{10} allowed us to understand the useful metrics for evaluating data quality.
- Q_{11} and Q_{12} allowed us to understand the most relevant data features.
- Q_{13} allowed us to understand if we had not considered some relevant aspects.

4.2 Data Collection

After a preliminary analysis of the existing project, we observed that this project uses a very small dataset, composed only of 170 images. Generally, a deep learning (DL) model needs a lot of training data for good performance. That is why we researched new data however, it is also for defining a dataset to train and test our model. On **Kaggle**, we have found several datasets related to the skin cancer problem. We chose two datasets with which we tested the selected approach and trained our models:

- The first one, available at this link, has 10000 images divided into two folders, namely test and train. Each folder contains two sub-folders named *benign* and *malignant* that contains the negative and positive instances respectively. The number of samples for the negative class in the entire dataset is 5500, instead for the positive class is 5105.
- The second one is available at this link. Also in this case, there are 10000 images, but they are classified into more category groups. Moreover, there are two folders that contain all the images and `csv` file named *HAM10000_metadata* that report the matching between the image ID and the related category of problem. The categories are:

- **akiec**, Actinic keratoses and intraepithelial carcinoma / Bowen's disease. There are 327 samples of this category in the entire dataset;
- **bcc**, basal cell carcinoma. There are 514 samples of this category in the entire dataset;
- **bkl**, benign keratosis-like lesions such as solar lentigines / seborrheic keratoses and lichen-planus like keratoses. There are 1099 samples of this category in the entire dataset;
- **df**, dermatofibroma. There are 115 samples of this category in the entire dataset;
- **mel**, melanoma. There are 1113 samples of this category in the entire dataset;
- **nv**, melanocytic nevi. There are 6705 samples of this category in the entire dataset;
- **vasc**, vascular lesions such as angiomas, angiokeratomas, pyogenic granulomas and haemorrhage. There are 142 samples of this category in the entire dataset.

Therefore, in this case, we need to collapse these different categories into two classes. We did this after we conducted some research to verify if each class is benign or malignant.

4.3 GA FOR MODEL IMPROVEMENT

Genetic Algorithms are based on biological evolution (i.e. Darwins's theory of evolution) and they are research algorithms for solving optimization problems. GA is capable of evolving a population of individuals iteratively. Each step produces a new generation of individuals improved over a fitness function. It repeats this improvement until a stop criterion is not verified. There are three operations that allow us to enhance the individuals, namely selection, crossover and mutation. Each individual is composed by m features called **genes**.
As shown in Figure 4 the lifecycle of GA is based on six phases:

(1) **Initialization**: in this phase, an initial population is defined, in most cases, randomly;
(2) **Selection**: in this phase, a set of parents has chosen from the previous generation based on fitness values. Those individuals are going to compose the new generation and they are going to be used to generate new individuals.
(3) **Crossover**: this phase represents the genetic operator used to combine the genetic information of two parents to generate new individuals;
(4) **Mutation**: this phase represents the genetic operator used to maintain genetic diversity, just like in biological mutation;
(5) **Evaluation**: in this phase, the modified population is assessed against the fitness function;
(6) **Replacement**: in this phase, the obtained population replaces the old one and a new iteration starts; Steps 2-6 are iterated until a stop criterion is not verified.

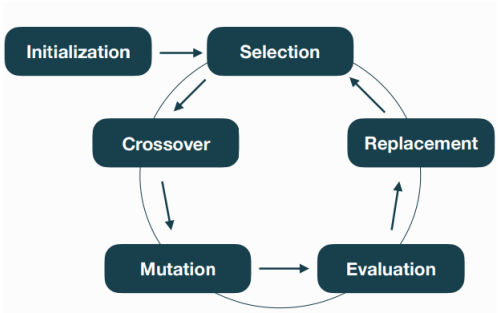

Figure 4: GA lifecycle

According to *Bochinski et al.*[2], the optimal choice of hyperparameters is one of the major challenges when applying CNN-based methods. Furthermore, since a broad range of CNN network structures yields good results, it stays unclear if a current solution obtains optimal or near-optimal configurations or if different structures can improve previous results. Genetic Algorithms can deal with these challenging conditions. That is why to improve the architecture and hyperparameter configuration of the model we used Genetic Algorithms (GA).

4.3.1 Genetic Algorithm Configuration.
As mentioned in Section 4, we developed three genetic algorithms focused on different aspects, namely hyperparameters of the training algorithm and network architecture. The objective function, the fitness function, the stop criterion and the initial population size of all GA versions that we developed are the same. The **objective function** is defined as follows:

$$max\ w * accuracy + (1 - w) * recall$$

w is a weighting factor that determines the trade-off between accuracy and recall; it is a value between 0 and 1. The **fitness function** is equal to the objective function and each individual has been assessed as follows:

$$f(n) = \frac{f(n)}{\sum_{n_i \in N - \{n\}} f(n_i)}$$

where N is the individuals set and n is an individual. In this way, we could evaluate an individual based on the evaluation of the entire population. The **initial population** consists of 100 random individuals. The **selection technique** that we used for all GA versions is K-way Tournament Selection, where $K = 30$ and $M = 20$. The **mutation technique** for all GA versions is Random Resetting. In the end, the **stop criterion** is the same used by Di Biasi et al[3].

Regarding the GA for the hyperparameters:
The goal of this version is to find an optimal training algorithm configuration. Therefore, we defined a set of individuals that represent a possible training algorithm configuration. We defined each individual as a vector $N = \{C_1, ..., C_m\}$ of m genes ($m = 5$). The position of a gene in the genome represents a certain hyperparameter. In particular, the hyperparameters that we chose for tuning are:

(1) **Learning rate**: it refers to the step size used during optimization to update the model weights. Therefore, it represents how much model weights must be modified to improve

model performance. The range that we chose for this parameter is between 0.001 and 0.1. This parameter has represented by the first gene in an individual genome;
(2) **Number of epochs**: it represents the number of epochs to train the model. An epoch is an iteration over the entire dataset. The range that we chose for this parameter is between 20 and 50. This parameter has represented by the second gene in an individual genome;
(3) **Batch size**: it represents the number of samples that a batch contains and the range that we chose for this parameter is the following: {65, 128, 256}. It has represented by the third gene in an individual genome;
(4) **Class_weight**: it is a dictionary mapping class indices (integers) to a weight (float) value, used for weighting the loss function (during training only). It is useful to tell the model to "pay more attention" to samples from an under-represented class. In our study, we have two classes, therefore we defined two ranges, one for each class. For the positive class we chose a range between 0.5 and 1, instead, for the negative class we chose a range between 0.4 and 0.7. This parameter has represented by the last two genes in an individual genome;

Hyperparameters which are not on the list have been set with the default value. The **crossover technique** that we used in this version is the One-Point Crossover.

Regarding the GA for the network architecture:
The goal of this version is to find an optimal CNN network structure. Therefore, we defined a set of individuals that represent the network architecture. We defined each individual as a vector $N = \{L_1, ..., L_m\}$ of m genes. Each gene represents a network layer. For each individual, we did not represent the first and the last network layer because the first one was fixed to the Input Layer, and the second one was fixed to the Output Layer for binary classification. In particular, a gene can be one of these layer types:

(1) **Preprocessing layer**: These layer types can increase the image quality to improve model performance. We will choose other preprocessing layer types based on survey results.
  (a) **RandomContrast Layer**: this layer will randomly adjust the contrast of an image by a random factor. Contrast is adjusted independently for each channel of each image during training;
  (b) **RandomBrightness Layer**: it randomly adjusts brightness during training;
(2) **Convolutoinal layer (Conv2D)**
(3) **Layer activation**: this layer type represents an activation function that is applied after a convolutional layer or fully connected layer or other layer types. It is applied when the other layers do not have an activation function defined;
(4) **Padding**: it can resolve the loss of information that might exist on the border of the image by adding an extra pixel to the input. This allows us to manage the output size;
(5) **Dropout**: it randomly sets input units to 0 with a frequency of rate at each step during training time, which helps prevent overfitting;
(6) **Pooling**: it down-sampling the input representation to reduce the complexity for further layers;
(7) **Fully connected layer**

The parameters of the layers have been set empirically. The **crossover technique** that we used in this version is the Two-Point Crossover.

Regarding the GA that combine the previous versions:
The goal of this version is to find an optimal CNN network structure and an optimal training algorithm configuration. Therefore, we defined a set of individuals that represent both the network architecture and training algorithm configuration. We defined each individual as a vector $N = \{C_1, C_2, C_3, C_4, C_5, L_1, ..., L_m\}$ of $m + 5$ genes. The first five genes represent the same hyperparameters of the first GA version; instead, the other m genes represent network layers. The values range of the first five genes is the same as the first GA version and the values range of the other m genes is the same as the second GA version. The **crossover technique** that we used in this version is the Two-Point Crossover.

4.4 Data Analysis

The methodical steps described in Section 4 allowed us to address the research questions, namely:

- The survey helped us to address the following research questions: RQ1, RQ2 and RQ4;
- The image processing techniques helped us to improve our model performance. These techniques have been chosen based on the survey results and helped us to address the research questions RQ3 and RQ4;
- The new found datasets, helped us to understand the limit of the chosen approach, therefore to address the research question RQ1. Moreover, we used these datasets for training and testing our model;
- Genetic algorithms helped us to improve the architecture and the configuration of the training algorithm hyperparameters, therefore to address the research question RQ3.

Afterward we obtained the results of all methodical steps, we evaluated them using different techniques. Regarding the survey results, we used these techniques:

- The survey results with an integer value have been analyzed with statistical techniques, namely mean, median and the box plot of the distribution;
- The survey results with a text value have been analyzed with thematic analysis in order to find common patterns.

We evaluated our datasets based on the metrics that we obtained from the survey results. In the end, regarding our model, we evaluated it based on these metrics: accuracy, recall, precision and F1-score. We assigned more weight to the accuracy and the recall because we must reduce the number of false negatives (indeed, high recall indicates a low number of false negatives) and create a capable model to predict each class correctly.

REFERENCES

[1] Saad Albawi, Tareq Abed Mohammed, and Saad Al-Zawi. 2017. Understanding of a convolutional neural network. In *2017 international conference on engineering and technology (ICET)*. Ieee, 1–6.
[2] Erik Bochinski, Tobias Senst, and Thomas Sikora. 2017. Hyper-parameter optimization for convolutional neural network committees based on evolutionary algorithms. In *2017 IEEE international conference on image processing (ICIP)*. IEEE, 3924–3928.

[3] Luigi Di Biasi, Alessia Auriemma Citarella, Fabiola De Marco, Michele Risi, Genoveffa Tortora, and Stefano Piotto. 2023. Exploration of Genetic Algorithms and CNN for Melanoma Classification. In *Artificial Life and Evolutionary Computation: 15th Italian Workshop, WIVACE 2021, Winterthur, Switzerland, September 15–17, 2021, Revised Selected Papers*. Springer, 135–138.

[4] Thomas P Quinn, Manisha Senadeera, Stephan Jacobs, Simon Coghlan, and Vuong Le. 2020. *Trust and medical AI: the challenges we face and the expertise needed to overcome them.* https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7973477/

[5] Dirk Schadendorf, David E Fisher, Claus Garbe, Jeffrey E Gershenwald, Jean-Jacques Grob, Allan Halpern, Meenhard Herlyn, Michael A Marchetti, Grant McArthur, Antoni Ribas, et al. 2015. Melanoma. *Nature reviews Disease primers* 1, 1 (2015), 1–20.

[6] Kun-Hsing Yu, Andrew L Beam, and Isaac S Kohane. 2018. Artificial intelligence in healthcare. *Nature biomedical engineering* 2, 10 (2018), 719–731.