Communication

NAILS: Normalized Artificial Intelligence Labeling Sensor for Self-care Health

Livio Tenze and Enrique Canessa *

Posted Date: 30 October 2024

doi: 10.20944/preprints202410.2375.v1

Keywords: region-based image segmentation; machine learning algorithm; fingernails color analysis; early disease detection

Communication

NAILS: Normalized Artificial Intelligence Labeling Sensor for Self-Care Health

Livio Tenze and Enrique Canessa *

SciFabLab, The Abdus Salam International Centre for Theoretical Physics (ICTP), 34151 Trieste, Italy
* Correspondence: canessae@ictp.it

Abstract: Visual examination of nails can reflect the human health status. Diseases such as nutritive imbalances and skin diseases can be identified by looking the colors around the plate part of the nails. We present the AI-based NAILS method to detect fingernails through segmentation and labeling. We built a generalized model to identify single fingers, their fingernail plates and retrieve their color data. A pre-trained deep learning Neural Network algorithm containing different images of fingernails as reference frames is used to identify different fingernails edges and the color areas around them in real-time. The novel aspect of our NAILS sensor model is that it allows to analyze –via a GUI and the use of a HD webcam, the tiniest dynamical signals from changes in fingernail colors. This feature could be used to self-extract primary signs of diseases in humans. NAILS binaries: https://github.com/canessae/nails

Keywords: region-based image segmentation; machine learning algorithm; fingernails color analysis; early disease detection

1. Introduction

The examination of nails in the limbs of a person plays an important role in medicine. In human beings, nails are the farthest part of the body from the heart, and are the last part to receive oxygen. Hence, an attentive observation of nails can be one initial method to check the physiological condition of the human body. Diseases can be identified by looking particularly at the lunula and plate parts of the nail. Dependent upon their shape, texture and color, these analyses can predict early symptoms of diseases and can be indicators of, e.g., nutritive imbalances, skin diseases, etc [1–4]. In particular, the ratio lunula to the plate nail, and the tiniest differences in nails colors, can reflect the ongoing human health status. Therefore, it is relevant to develop a first self-care health sensor method to detect color nails changes as alternative to dermatological exams and scanning tests.

Advances in Artificial Intelligence (AI) –including fast Convolutional Neural Networks (CNN) object detection models and machine learning, allow to perform accurate tracking using only hands' images taken from a HD webcam. Combining these advancements with libraries like TensorFlow, AI provides an efficient framework for deploying interactive algorithms for object detection and hand-tracking. With the use of sufficiently large data-sets of diverse hand postures, CNN can provide an alternative approach to train models that perform reasonably well when addressing challenges such as sharp changes in lighting conditions, noisy environments, different hand viewpoints and the account of fingers occlusion when carrying out measurements.

Several prototypes have been proposed to detect fingernails and hand gesture interactions from computer vision [5,6]. One way is to develop pre-trained hand detection model by CNN and Deep Neural Networks. Different images in the given data sets, containing particular hands gestures as reference frames, provide to identify different input finger edges and areas around the finger plates. In training a hand detector sensor, multi-processes are needed to assemble a data set, cleaning and splitting into training/test/validation partitions and generating a display interface. This is built using data from unique hand data sets available on the Internet [7,8]. These data sets usually contain pixel level annotations where hands are located across hundreds of high quality images. Images are captured from egocentric and alternative view across different indoor and outdoor environments and activities using hands positions such playing games, etc. Hence fast CNN or deep learning algorithms with pre-trained models are an attractive candidate for real-time fingernails detection and tracking applications, as the one we report in this paper.

In this work we introduce an AI-based sensing method to detect fingernails through CNN from which we detect nails plates and colors through segmentation and labeling. We built this sensor-like method from a generalized model to identify single fingernails and from it, we retrieve color nail plates data. A simple pre-trained CNN algorithm containing particular snapshots of fingernails as reference frames is used as example which allows to identify different input fingernails edges and nuances. The novel aspect of our Normalized Artificial Intelligence Labeling Sensor (NAILS) is that it allows to analyze tiniest dynamical signals from changes in nail plate colors through the use of a USB webcam and an easy-to-use Graphic User Interface (GUI). We also discuss how this ad-hoc GUI could be used for self-care health surveillance to extract primary signs of diseases rather quickly.

2. Previous Work

Some hands tracking and detection algorithms by computer vision are not very robust specially when these are based, for example, on extracting boundary features and backgrounds depending only on texture. This is also the case when distinguishing between hands and background using just histograms or color maps. In fact, on unusual backgrounds where varied or poor lighting causes changes in skin color, or on situations in which the tracked hand becomes partially occluded, classical algorithms can easily lead to wrong results.

There are a few innovative and alternative AI-based, open source software available which allows to identify fingernails by applying CNN and image processing algorithms. One of these is the `nailtracking` [6]. This is a real-time nail-detection algorithm using CNN on TensorFlow. The deep learning framework or TensorFlow object detection allows to simplify the process of training a model for custom object detection. It allows, for instance, to draw a line when identifying the dynamics of a finger in action over a surface. The goal behind `nailtracking` is to provide an adaptable Python code to demonstrate how CNN can be applied to tracking hands having egocentric and alternative views. The different hand images in the given data sets contain particular gestures which are used as reference frames and lead to identify different finger edges.

Another available algorithm for nails segmentation using deep learning models is denoted as `nails_segmentation` [5]. For the segmentation part this algorithm employs the encoder-decoder structure `DeepLabV3+` [9], which allows to fast recover the object boundaries. `nails_segmentation` consists in four basic steps for data preparation, training, prediction and evaluation. On diverse public data sets test performances, this model can achieve higher than 80% without the need of post-processing. Furthermore, if there is a pre-trained model (denoted "`best_model.pth`"), new results can faster be retrieved from scratch using that given model. For these two reasons, `nails_segmentation` has been adoped as the AI-based algorithm to construct a GUI to easily identify plate areas and colors of fingernails by image processing in real-time, as discussed below.

3. NAILS Algorithm

The idea behind NAILS is to develop an algorithm aiming at detecting fingernails colors by image processing in real time by using the CNN `nails_segmentation` method [5]. We have found that this CNN performs better than alternative solutions because it is based on the nails segmentation using fast deep learning models. Training is performed using public data sets comprising fingernails images of individuals. Data set for nails segmentation, with images and annotated masks, can be found in [7,8]. Usually these masks are cleared up from the background noise which may appear because these are created using gray and white color ranges. A median blur is used to fill holes in the masks. The weights are applied to real images with reasonable good results. We classify the images on the basis of fingernails implemented following the flow diagram below.

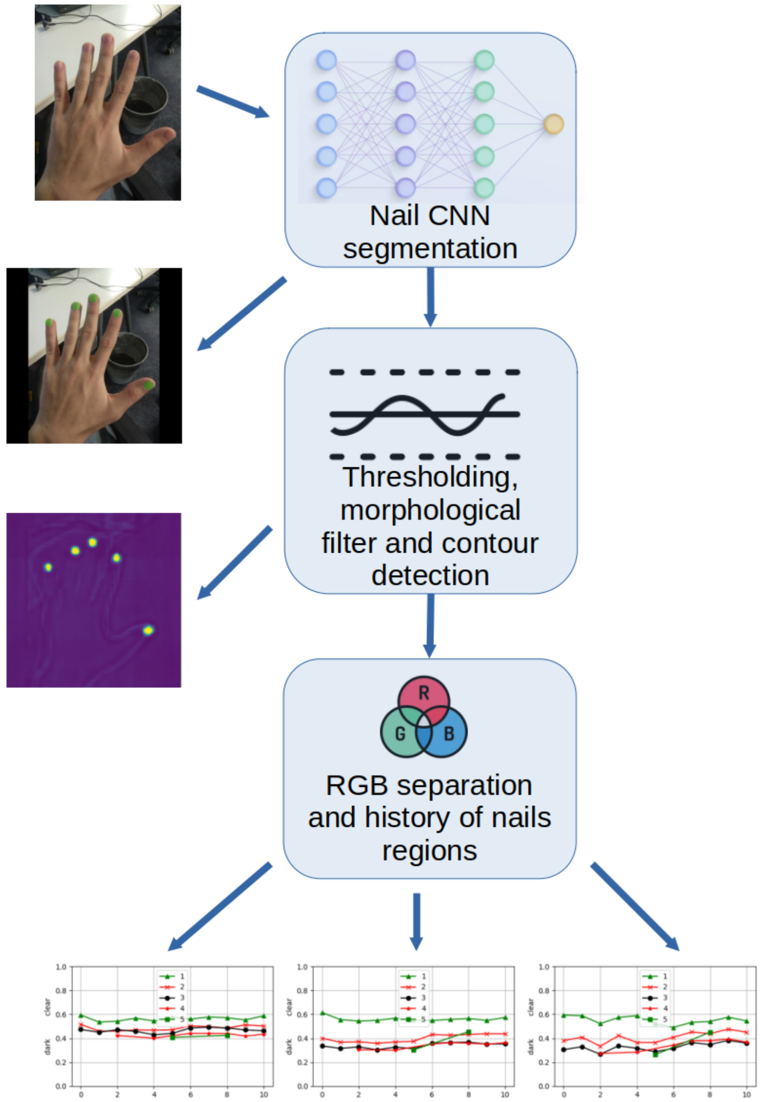

Nail CNN segmentation
Thresholding, morphological filter and contour detection
R
G
B
RGB separation and history of nails regions
clear
dark
1
2
3
4
5
0.0
0.2
0.4
0.6
0.8
1.0
0
2
4
6
8
10

Diagram: NAILS *sequence algorithm. The hand images are examples from the data set in [7].*

When the user gets a snapshot from the video source showing an open hand (subdivided in five equally rectangular areas separated by blue lines as in the input video on the left of Figure 1), the system:

- Search for, and processes, reference Black and White regions in order to normalize the image pixels (I_x) and therefore the color values of the nails. The algorithm then normalizes the R(ed), G(reen) and B(lue) values at pixel x according to the equation:

$$\eta_x = \frac{I_x - Black_x}{White_x - Black_x} \quad , \tag{1}$$

where x can be R, G or B region displayed on the left video.

- The referential Black value is evaluated taking into account the minimum value inside the small squared regions located at the four corners of the input video. The referential White value is evaluated taking into account the maximum value inside the small squared regions located at the three central areas of the input video.
- Then the CNN is fed by the acquired image in order to get the nails regions. CNN use the convolution layers (or layers that use specialized linear operations instead of the matrix multiplications) to extract features from the target objects. In our case the CNN from [5] has been adapted to run under Ubuntu O.S. This network performs better than alternative solutions because it is based on the segmentation concept. The segmentation is the process of dividing a digital image into different parts, in which each part consisting of homogeneous pixels, distinguishes the object or other information contained in the image. The network has been trained over the default data set and the trained weights are applied to real images with reasonable good results.
- The mask evaluated by the CNN network provides regions where nails are present.
- The mask is post-processed by thresholding and morphological filters to improve the previous detection to get sharp contours around the nails.
- Finally the system gets the RGB values from the nails regions (based on pixels values inside contours), normalizes the RGB values and stores the RGB normalized values.
- As illustrated in Figure 1, the RGB normalized values of the respective R,G,B curves for each fingernail identified by the algorithm are plotted in the bottom region of the GUI, where the RGB history is displayed.

The minimum hardware needed to test NAILS is any standard PC Computer *e.g.*, Intel Core i5, 64bit and at least 4G RAM, running a recent release of Linux O.S. (24.04 LTS or newer) having an internal webcam and Python installed.

In case of using an external USB webcam, the user needs to install the `v4l-utils` package, which can enumerate and detect many aspects of installed cameras. This class has been mainly added to NAILS in order to implement the camera combo box to allow the user to change the current acquisition camera using the menu in Figure 2. Optional hardware that can be used is an external USB webcam (with optical zoom).

4. Fingernails Color Detection via NAILS

The GUI interface of the NAILS package shown in Figure 1 is based on the `tkinter` Python library. Some other libraries have been integrated in order to add plot history (with the `matplotlib` and `numpy` libraries being used), to show the real-time and snapshots acquired from a USB webcam (`opencv-python`). Some classes have been developed in order to solve other tasks such as videocapture pause and restart, complex processing of the acquired images and so on.

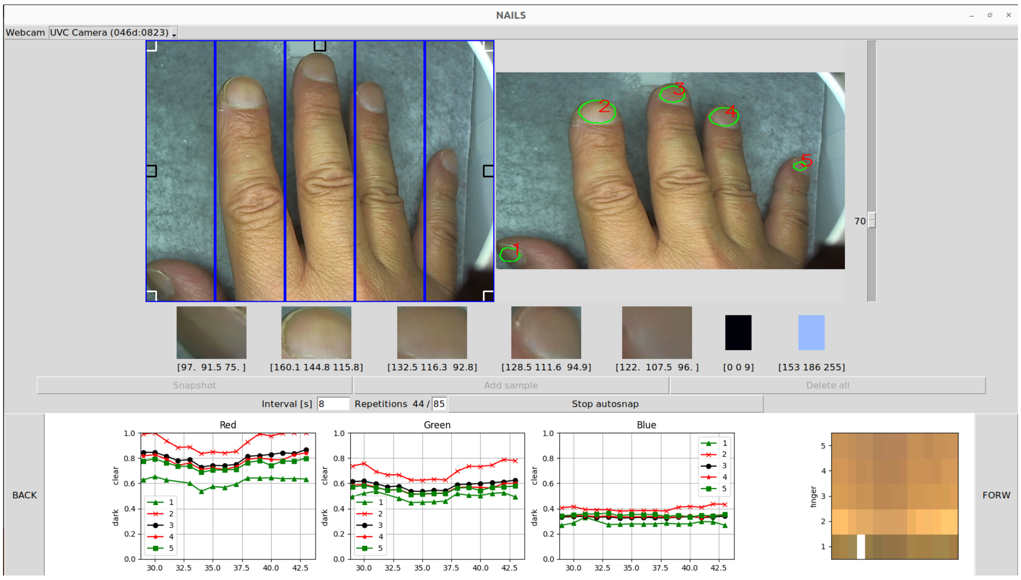


Figure 1. *GUI for the Normalized Artificial Intelligence Labeling Sensor NAILS aiming to identify hidden signals from fingernails.*

To start using the NAILS GUI, the acquisition webcam from which to get the video image of the hand (input video on the left of Figure 1) needs to be selected. The `CameraSearch` library was implemented in order to enumerate and to get info about the connected webcams: no one library completely functional was found from the open source solutions available in the Internet. In order to make the `CameraSearch` library working, it is necessary to install the `v4l-utils` package, hence our current GUI solution is currently not portable to Windows or MacOS systems. It can enumerate and detect many aspects of installed cameras. This class has been mainly used to implement the camera combo box to allow to select the acquisition camera as illustrated in Figure 2. In this figure, the menu listing one internal webcam and a second UVC webcam connected to the PC via USB is shown.

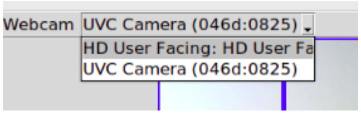


Figure 2. *Selection webcam menu.*

After positioning each finger in one of the five equally rectangular areas separated by blue lines in the GUI input video, used as guidance, there are two modalities to get the nails regions through the NN method used by NAILS. As illustrated in Figure 3, one modality is to acquire single images for the nails regions by selecting and pressing the "`Snapshot`" button or just pressing the "`Enter key`". When selecting the "`Snapshot`" single button in the GUI, one can "`add sample`" or press the "`Space key`" to plot the single RGB values estimated. After a few seconds the identified areas around the fingernails are displayed (output image on the right of Figure 1). The second modality, is to acquire a large temporal set of data at given intervals by selecting "`Start autosnap`".

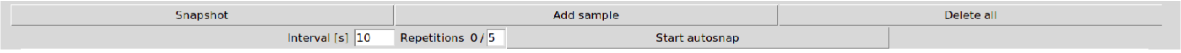


Figure 3. *Modalities to acquire (single or multiple) images for the fingernails regions.*

The areas around the fingernails can be regulated in size by the cursor appearing on the right of the GUI as shown in Figure 4 as example. In all cases the plotted values surrounding these areas, correspond to a mean value obtained for each single fingernail region identified. By selecting "`Delete all`" all recorded data is cancel.

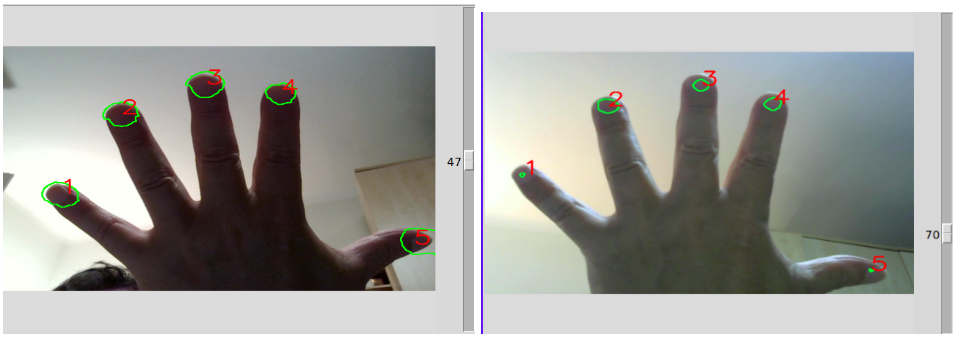


Figure 4. *Fingernail areas regulated in size by the cursor on the right of the GUI.*

The NAILS algorithm then gets all the actual RGB color of the (five) fingernail regions as shown in Figure 5. These single colors are based on the theory that all visible colors can be recreated using the primary additive colors of Red, Green and Blue assigning a value in the range of 0–255. When these values are combined in different amounts, fingernails colors are produced. For example: (0,0,0) is black and (255,255,255) is white.

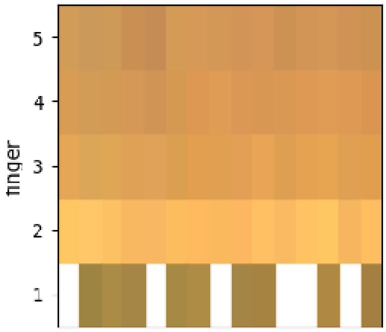


Figure 5. *The visible RGB colors of the five fingernail regions being identified at each measurement. The empty white values corresponds to missing measures by the CNN algorithm, usually due to the occlusion and position of the thump with time. The observed nail color range is usually similar to the skin color and depends on luminosity.*

As can be seen in Figure 6, the GUI shows in real-time the three normalized R-G-B values identified by the algorithm for each fingernail region, normalized using Eq.(1). The R-G-B recorded values during a period of time (configured at given intervals and repetitions within the GUI) can be visually compared by using the `Forward` and `Backwards` buttons on both edges of the plot.

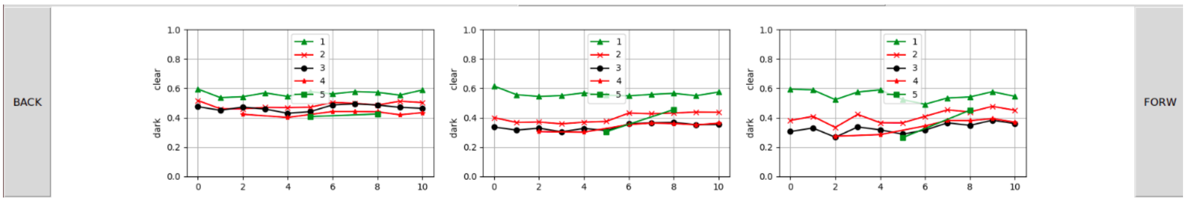


Figure 6. *Red, Green and Blue plots of the mean color values for each finger.*

5. Discussion

Our project aims at detecting symptoms of various diseases in their early stages. NAILS may help to associate pre-symptoms according to literature [1,10,11] since the color of (finger)nails may reflect the health status of an individual as for example those depicted in the Table 1.

Table 1. Nail color and Disease symptoms.

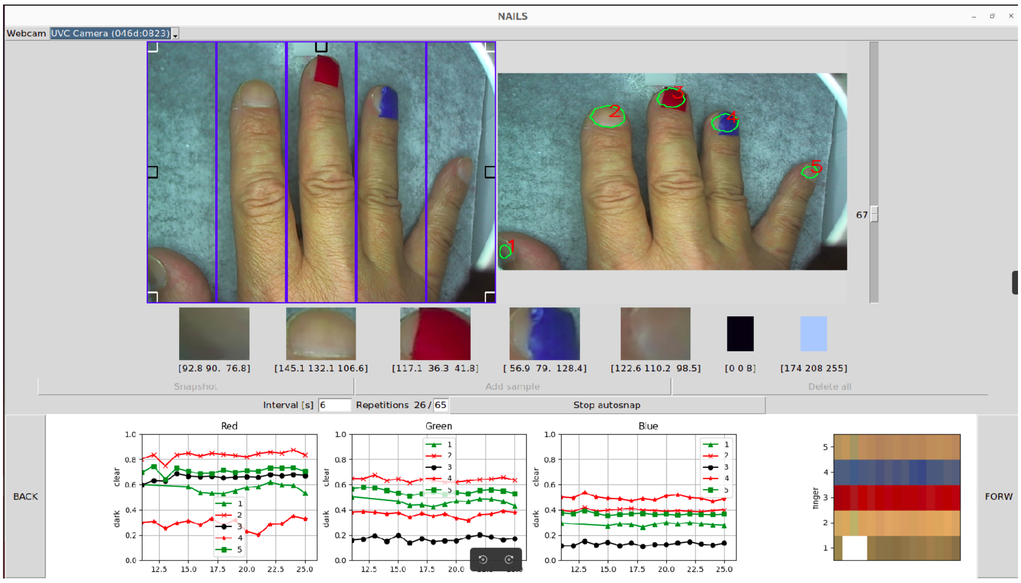
Nail color	(RGB)	Examples of possible diseases
Yellow	(255,255,0)	diabetes or psoriasis, lung disease
Green	(0,255,0)	Allergies to cleaning agents, localized fungal infection
Brown	(165,42,42)	Arsenic or copper poisoning, nicotine
Red	(255,0,0)	Injury, splinter hemorrhage, high blood pressure
White	(255,255,255)	protein deficiency, anemia
Purple	(128,0,128)	Oxygen deprivation, circulatory problems

There are a few issues leading to noticeable performance increases on the results observed through the AI-based NAILS algorithm and the CNN advances to make these results faster [12]. The CNN used is a type of neural network architecture with many sub sampling layers capable of performing object detection with high-level accuracy [13]. Each layer in this network is able to extract the target object features. NAILS can be used for real-time processing. The time that the NAILS algorithm is able to process a measurement is less than 2 seconds (without GPU cores).

The size of the reading images from the webcam is set by default to 800x800 px to avoid slowing down the program in real-time. Keeping these input images small allow to increase the video fps without losing significant accuracy. The observations are made fast because a pre-trained model (namely, "`best_model.pth`") is given with the sources of NAILS, which is trained on Ubuntu O.S. It took around two hours to generate the NN on a i9 CPU machine.

Lighting conditions on the system can cause sharp changes in fingernail color. The partial occlusion of one tracked finger, or the fingernail area is too tiny to detect (e.g., half thumb), can affect detection results and not be able to retrieve useful color data for that finger. For these reasons, it is advisable to use a closed system (box) to insert the hand to make more precise measurements in a luminous controlled environment (e.g., resembling the ancient "Mouth of Truth" attraction in Roma, Italy). This allows to have every finger visible and separated from the background.

NAILS does not deal with the segmentation of specific parts of the nail like lunula. It uses a simple image processing method to segment nails from given images. NAILS identifies and decouple visible RGB colors of fingernail regions at each measurement as shown in Figures 5 and 6. An example for a critical assessment of the NAILS behavior (e.g., when forcing red and blue fingernail colors to picture effective changes) is display in Figure 7.



Webcam UVC Camera (046d:0823)

[92.8 90. 76.8] [145.1 132.1 106.6] [117.1 36.3 41.8] [56.9 79. 128.4] [122.6 110.2 98.5] [0 0 8] [174 208 255]

Interval [s] 6 Repetitions 26 / 65

Figure 7. *Critical assessment of the NAILS behavior by forcing red and blue fingernail colors.*

Our human fingernail segmentation with model training algorithm can then be used as a first self-diagnostic tool. NAILS overcomes the limitations of the human eye resolution and provides an objective evaluation of the nails color. In all cases, however, a definitive diagnosis is always up to the Dermatologist specialist since laboratory tests are always necessary. Hidden signals in the discoloration of the nails using AI together with the observation of other symptoms and physical examinations, could be useful for diagnosing a specific disease specially in remote areas or when standard testings by accurate reading machines are not being used or are not always available.

6. Conclusion

We have introduced NAILS, an AI-based, non-invasive, sensor-like method for inspection of fingernails through segmentation and labeling. We built a generalized model to identify single fingers, plates and color data. By construction, a pre-trained deep learning neural network containing different images of fingernails has been used as reference frames to identify fingernails edges and color areas in real-time. The RGB average value of the input finger nail color can be used to classify some diseases.

The main aspect of NAILS is that it allows to analyze –via a simple GUI and a USB webcam, dynamical tiniest changes in fingernail colors. This feature could be used as an indicator of human health status. Generally healthy fingernails are shiny and smooth in appearance.

We believe NAILS to open the door to run on different configurations. The NAILS algorithm can also be embedded in a Raspberry Pi with Alpine Linux installed. In addition, NAILS could also be adapted to run on mobile devices. All of these directions have the potential to be explored in future releases.

7. NAILS Binaries

Further information, binaries, papers, presentations, manuals, or to report bugs, can be found at https://github.com/canessae/nails

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Folle U.; Fenzl P.; Fagni F. DeepNAPSI multi-reader nail psoriasis prediction using deep learning *Scientific Reports* **2023**, *13*, 5329. doi: 10.1038/s41598-023-32440-8
2. Sharma V.; Ramaiya M. Nail Color and Texture Analysis for Disease Detection. *Int. J. Bio-Sci. and Bio-Tech.* **2015**, *7*, 351–358. doi: 10.14257/ijbsbt.2015.7.5.34
3. Abdulhadi J.; Al-Dujaili A.; Humaidi A.J.; Fadhel M.A-R. Human Nail Diseases Classification Based on Transfer Learning. *ICIC Express Letters* **2021**, *15*, 1271–1282. doi: 0.24507/icicel.15.12.1271
4. Kumar D.S.; Sherly, J.D.; Priyadharshini, S.I. Disease Detection Based on Nail Color Analysis Using Image Processing. 1st Inter. Conf. Comp. Sci. and Tech. (ICCST), Chennai, India, **2022**, 1–5. doi: 10.1109/ICCST55948.2022.10040425.
5. `nails_segmentation`: Nails segmentation using deep learning models. Sources available online: https://github.com/ademakdogan/nails_segmentation (accessed on 20 Oct 2024)
6. `nailtracking`: Real-time Nail-Detection using Neural Networks (SSD) on Tensorflow. Sources available online: https://github.com/toddwyl/nailtracking (accessed on 20 Oct 2024)
7. Kaggle dataset for nails segmentation. Sources available online: https://www.kaggle.com/datasets/vpapenko/nails-segmentation (accessed on 20 Oct 2024)
8. Small dataset for nails segmentation. Sources available online: https://github.com/vpapenko/nails-segmentation-dataset (accessed on 20 Oct 2024)
9. Chen LC.; Zhu Y.; Papandreou G. et al. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. *Lecture Notes in Comp. Sci.* **2018** *11211*. Springer, Cham. Available online: https://doi.org/10.1007/978-3-030-01234-2_49 (accessed on 20 Oct 2024)

10. Rasminsky A. What These 8 Fingernail Signs Say About Your Health. Healthline Newsletter July 31, **2024**. https://www.healthline.com/health/beauty-skin-care/healthy-nails (accessed on 20 Oct 2024)
11. Klonoff D.C. What Do Your Fingernails Say About You? Can They Indicate That You Have Diabetes? *J. Diabetes Sci. and Tech.* **2015**, *9*, 1167–1169. doi: 10.1177/1932296815608980
12. Alghamdi M.; Angelov P.; Lopez Pellicer A. Person identification from fingernails and knuckles images using deep learning features and the Bray-Curtis similarity measure. *Neurocomputing* **2022**, *513*, 83–93. doi: 10.1016/j.neucom.2022.09.123
13. Nugroho B.; Mumpuni R.; Munir M.S. Performance of the CNN Method for Identifying Health Conditions Based on Nail Images. *Nusantara Sci. and Tech. Proceed.* **2023**, *23*, 658–664. doi: 10.11594/nstp.2023.33106

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.