Telehealth Ecosystems in Practice
M. Giacomini et al. (Eds.)

doi:10.3233/SHTI230744

A Custom Image Recognition Model in the Cloud for Skin Diseases Identification

Mircea-Aurelian DAN[a,1], Stelian NICOLA[a] and Lacramioara STOICU-TIVADAR[a]
[a] *Department of Automation and Applied Informatics, Politehnica University Timisoara, Romania*
ORCiD ID: Mircea-Aurelian DAN https://orcid.org/0009-0001-5755-6451
ORCiD ID: Stelian NICOLA *https://orcid.org/0009-0002-4878-6746*
ORCiD ID: Lacramioara STOICU-TIVADAR https://orcid.org/0000-0001-9700-7732

Abstract. In the context of global warming and increasing exposure to UV radiation, skin diseases are becoming more prevalent. Some of the most widespread skin conditions are solar lentigo and actinic keratosis. In this paper, we propose a technical approach related to the use of Azure Custom Vision services to classify these two conditions. The main advantage of using this service is the computational power offered by Azure. Additionally, generating a convolutional neural network model does not require a large dataset to achieve a good performance. For training the model, we used a dataset of 600 images from the ISIC database. The limitations of these approaches are imposed by the manual image labeling part that needs to be performed. As a result, we provide a trained model on a series of images that can be used for classifying images related to these two conditions. The performance of our neural network on the pre-trained images is 94%.

Keywords. custom image recognition, models, teledermatology

1. Introduction

Skin diseases are a broad category of conditions that affect the skin, specifically the outer layer of the skin. Being the largest external organ of the human body, it is exposed to the most dangers. Through the nerve centers located in the skin, we can feel heat or cold. The greatest danger that the skin is exposed to on a daily basis is represented by the sun's UV radiation. Solar lentigo is a skin condition that occurs as a result of excessive sun exposure. It is characterized by dark brown or black pigmented spots on the areas most exposed to the sun. The spots have a well-defined shape and are flat [1]. Actinic keratosis is a pre-cancerous skin lesion that usually appears on sun-exposed areas. It is caused by the accumulation of dead/damaged cells in the upper layer of the skin due to repeated and prolonged sun exposure. The spots have a well-defined shape, are dry, rough, and have a red, brown, or gray color [2].

In the specialized literature, numerous concerns of researchers have been identified in this area. Thus, Chakroborty T. and Mahmud F. [3] propose a classification model for various skin diseases by considering texture characteristics and balancing the dataset

[1] Corresponding Author: Mircea-Aurelian Dan, Politehnica University Timisoara, P-ta Victoriei no. 2, Timisoara, Romania, E-mail: mircea.dan@student.upt.ro.

describing the images with corresponding patches for each condition. The main diseases that are attempted to be classified are solar lentigo and lentigo simplex. The models used in disease classification are KNN (K-nearest neighbors) and SVM (Support Vector Machine). The best performances achieved, in terms of F-measure, were 0.869 for the SVM model and 0.847 for the KNN model. Another approach in skin disease classification can be found in Albahar M. A. [4]. He proposes a prediction system for classifying benign or malignant skin lesions using a novel regularization technique. The classifier for these lesions is binary, and the proposed model achieved an average accuracy of 97.49%.

The classification of three stages of skin cancer: solar lentigo, lentigo maligna, and lentigo maligna melanoma is proposed by Thungprue N. et al. [5]. They propose an innovative solution through semi-supervised learning of CNN (Convolutional Neural Network) based on VGG-16 and VGG-19. The accuracy achieved for supervised transfer learning was 92% for VGG-16 and 98% for VGG-19. Another model used in classifying images with skin conditions is CART (decision trees classifier). It is used for detecting the potential of solar lentigo and melanoma. The accuracy achieved by this model in classifying the two diseases was 97.5%. The CART model achieved these performances on a training set of 60 data points [6].

Another approach used for detecting the potential of skin cancer is addressed by Martiano H. F. et al. [7]. They propose an expert system for Android that aids in the detection of potential cancer: lentigo melanoma and nodular melanoma. In the mobile application, users are presented with a series of 10 questions aimed at detecting the likelihood of having the disease. Among the questions asked are: "Do you have family members with skin cancer?", "Do you have a nevus exposed to sunlight?", "Where is the nevus located?". The expert system also takes into account data obtained from processing images of pigmented patches on the skin. This study has shown that more than 60% of users of this application were advised to consult a specialist doctor. All these studies mostly use images from open access databases. ISIC (International Skin Imaging Collaboration) is among the most well-known open databases [8]. Researchers have at their disposal over 68,000 images of various skin conditions.

Another common aspect of these research studies is the development environment for the scripts aimed at image classification. We have identified MATLAB and the Python language as the most commonly used environments.

Regarding the training of neural networks, the computational power of local machines/computers is utilized. In the specialized literature, we have observed several approaches where Cloud services are used. These services provide automated learning algorithms that run on specialized hardware components to deliver high-performance. One of the newest services specialized in this field is provided by Microsoft: Azure Cognitive Services. The Cloud servers in Azure implement the application programming interface (API) of the image recognition application within the Vision category of these services. They leverage Cloud servers to perform the algorithm's recognition process. The solution offered by Azure Cognitive Services eliminates any real-time differences as everything runs in a chained manner. The main advantage of this solution is its execution speed, which does not compromise detection accuracy. The drawback is its dependency on the internet [9].

Azure Custom Vision, as part of Azure Cognitive Services, has started to be used in object/image detection. In the specialized literature, it is used for detecting food spoilage. As observed in [10], achieving high performance in food spoilage detection does not require a large number of images. Researchers achieved a detection precision of 85% on a set of 1000 training images. Furthermore, this service can be used to detect potential environmental pollution that can directly affect human health [11]. The algorithm proposed by Almeer S. et al. achieved a precision of 90.5%. The same Azure service was used to train the neural network model. Various software architectures are used in machine learning for improved classifier performance [13].

In this paper, we propose the use of Azure Custom Vision (ACV) service for training a convolutional neural network model aimed at detecting skin conditions: solar lentigo and actinic keratosis. Similar to the scientific papers mentioned above, we will use images provided by the open access platform of ISIC. The paper is divided into 3 chapters. The first chapter presents the main purpose of the paper and highlights the most remarkable results from the state of the art. The Methods and Tools chapter outlines the main steps to be taken in labeling, training, and testing the model. Additionally, it presents the topology of the neural network generated through CV. The Results and Conclusions chapter showcases the key outcomes on metrics such as accuracy, recall, and mean average precision of the constructed neural network. Perspectives are also discussed in this chapter.

2. Methods and Tools

In this chapter we will present the main tools we use in classifying the two mentioned skin conditions and which are the main steps that must be followed to use the service offered by ACV. The labeling, training and testing process is explained in the end of this chapter.

2.1. Labeling, Training, and Testing Process

For labeling images containing the skin conditions solar lentigo and actinic keratosis, we used the classification component of ACV, which allows uploading an image to the platform and labeling it based on the defined tag. Each tag we define represents a class of skin condition. We identified two classes named after the respective conditions: solar lentigo and actinic keratosis. After labeling all the images, the training process begins. There are two options for this process: quick training and advanced training. In advanced training, the maximum training time can be set (up to 96 hours). After the training process, the training metrics of the model, such as precision, recall, and mAP (mean Average Precision), can be observed on the ACV platform. If the performance of the trained model is satisfactory, we can export it in various formats: CoreML, TensorFlow, ONNX, Dockerfile, Vision AI Dev Kit, and OpenVino. If we want to test the model, we can directly test it from this platform using the Quick Test functionality. For that we can choose an image from our personal computer or from another database. In case multiple iterations of the model have been performed, we can select which iteration to use. As a result, we will receive the predicted class of the tested image, along with the class prediction accuracy. Figure 1 illustrates the steps of the process.

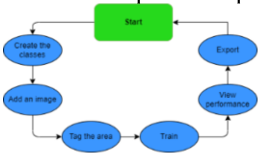

Figure 1. Workflow of ACV

2.2. Network topology

In Figure 2, we have the detailed topology of the neural network created in ACV. To display the topology of the created network, we opened the *.pb* (Protobuf) file in the Netron platform, an open-access platform.

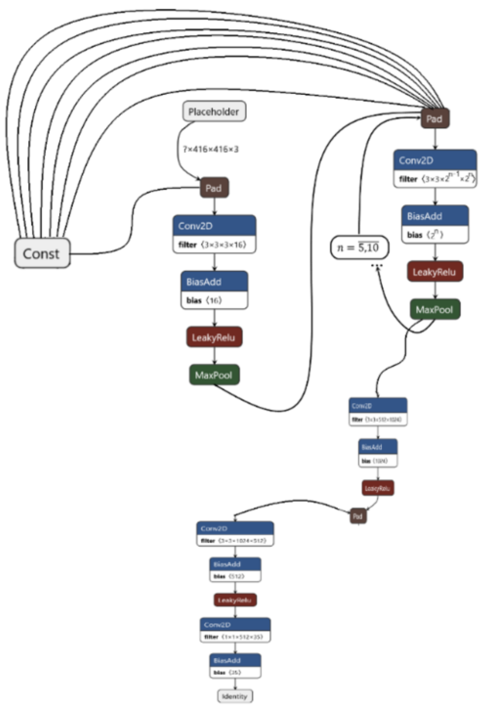

Figure 2. Network topology model.

In the Figure 2, the network was simplified having same pattern of: Conv2D, BiasAdd, LeakyReLu, MaxPool layers with different widths and channels. The full version of the network topology can be found to the our github repository [12]. In the first layer pattern, our input will consist of an unknown number of image samples having heights, widths and channels of 416x416x3. Within Conv2D layer, a kernel of size 3x3x3x16 will be applied followed by a bias of 16. LeakyReLu will stand for the activation function model. MaxPool will extract the maximum values of each kernel. For the next Conv2D layer, a padding will be applied within increased number of inputs and the calculated difference for the first layer pattern is 4176. This process will repeat 4 times until the layer pattern before output. At this time, the MaxPool layer will be missing. For the final steps, an other LeakyReLu will be applied on the last Conv2D, BiadAdd layers to generate the output.

3. Results and Conclusions

Because ACV is oriented on detecting objects in images without detecting skin diseases, for the first iteration of our trained model, we obtained the following performances:

Precision 60.2%, Recall 41.7%, mAP (mean average precision) 59.7%. The model was trained on a dataset of 300 images per skin disease. From the set of 300 images, the platform took 20% for testing, so the train was made with 240 images. We also tried the quick training facility from the tools, but on our test cases, the performance was lacking. However, when we made a second iteration having train time of 12 hours, the performance slightly differed, but we obtain a good probability when we try to make quick testing. For example, when we upload an image that was included in the train set, we get a probability of 94%.

The results are not the best, our future strategy would imply training on maximum amount of time from ACV. Also, a more variate set of images we might make new iterations. The images will be taken from different open access images databases.

References

[1] Andersen WK, Labadie BS, Bhawan J. Histopathology of solar lentigines of the face: A quantitative study, Journal of the American Academy of Dermatology, 2006, doi: 10.1016/s0190-9622(97)80224-1.

[2] Christensen RE, Elston DM, Worley B, McKenzie A, Noor A, Kang BY, Bahrami S, Robert T, Cerroni L, Elston C, Ferringer T, Hurley MY, Garton K, See Lee JS, Liu Y, Maize JC, McNiff JM, Rapini RP, Sangueza OP, Shea CR, Zhou C, Alam M, Dermatopathologic features of cutaneous squamous cell carcinoma and actinic keratosis: Consensus criteria and proposed reporting guidelines, Journal of the American Academy of Dermatology, Volume 88, Issue 6, 2023, doi: 10.1016/j.jaad.2022.12.057

[3] Chakroborty T and Mahmud F. An Approach for Classifying Skin Diseases Using Texture Analysis, 5th International Conference on Advances in Electrical Engineering (ICAEE), Dhaka, Bangladesh, 2019, pp. 16-20, doi: 10.1109/ICAEE48663.2019.8975547.

[4] Albahar MA. Skin Lesion Classification Using Convolutional Neural Network With Novel Regularizer, in IEEE Access, vol. 7, pp. 38306-38313, 2019, doi: 10.1109/ACCESS.2019.2906241.

[5] Thungprue N, Tamronganunsakul N, Hongchukiat M, Sumetpipat K and Leeboonngam T. Using Semi-supervised Transfer Learning for Classification of Solar Lentigo, Lentigo Maligna, and Lentigo Maligna Melanoma, 14th Biomedical Engineering International Conference (BMEiCON), Songkhla, Thailand, 2022, pp. 1-5, doi: 10.1109/BMEiCON56653.2022.10011586.

[6] Bimastro KZ, Purboyo TW, Setianingsih C and Murti MA. Potential Detection of Lentigo Maligna Melanoma on Solar Lentigines Image Based on Android, 4th International Conference on Information Technology, Information Systems and Electrical Engineering (ICITISEE), Yogyakarta, Indonesia, 2019, pp. 113-118.

[7] Martiano HF, Purboyo TW and Setianingsih C, Detection of Potential Skin Cancer Lentigo Maligna Melanoma and Nodular Melanoma with Expert System Using Variable-Centered Intelligent Rule System (VCIRS) Method, 6th International Conference on Instrumentation, Control, and Automation (ICA), Bandung, Indonesia, 2019, pp. 42-46.

[8] ISIC Archive, Jun. 2023, Available: https://www.isic-archive.com/.

[9] Vani Ch, Reddy GR. Development of internet of things and cloud based applications using Kivy for mobile and smartphone integration, Materials Today: Proceedings, 2021, ISSN 2214-7853, https://doi.org/10.1016/j.matpr.2021.01.753.

[10] Batugal CL. EyeSmell: Rice Spoilage Detection using Azure Custom Vision in Raspberry Pi 3, IEEE REGION 10 CONFERENCE (TENCON), Osaka, Japan, 2020, pp. 738-743, doi: 10.1109/TENCON50793.2020.9293915.

[11] Almeer S, Albalooshi F and Alhajeri A. Oil Spill Detection System in the Arabian Gulf Region: An Azure Machine-Learning Approach, International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT), Zallaq, Bahrain, 2021, pp. 418-422, doi: 10.1109/3ICT53449.2021.9581841.

[12] Github repository, https://github.com/MirceaAurelian/Azure-CustomVision-Article.

[13] Kleftakis S, Mavrogiorgou A, Zafeiropoulos N, Mavrogiorgos K, Kiourtis A and Kyriazis D. A Comparative Study of Monolithic and Microservices Architectures in Machine Learning Scenarios, IEEE International Conference on Computing (ICOCO), Kota Kinabalu, Malaysia, 2022, pp. 352-357, doi: 10.1109/ICOCO56118.2022.10031648.