A HYBRID DEEP TRANSFER LEARNING MODEL WITH MACHINE LEARNING METHODS FOR FACE MASK DETECTION IN THE ERA OF THE COVID-19 PANDEMIC

Syed Shabaaz Ali[1], Syed Sultan Mohiuddin Hussaini[2], Mrs.M.Shilpa[3]

B.E Student, Department of IT, Lords Institute of Engineering and Technology, Hyderabad

Assistant Professor, Department of IT, Lords Institute of Engineering and Technology, Hyderabad

shilpa.m@lords.ac.in

ABSTRACT: The coronavirus COVID-19 pandemic is causing a global health crisis. One of the effective protection methods is wearing a face mask in public areas according to the World Health Organization (WHO). In this paper, a hybrid model using deep and classical machine learning for face mask detection will be presented. The proposed model consists of two components. The first component is designed for feature extraction using Resnet50. While the second component is designed for the classification process of face masks using decision trees, Support Vector Machine (SVM), and ensemble algorithm. Three face masked datasets have been selected for investigation. The Three datasets are the Real-World Masked Face Dataset (RMFD), the Simulated Masked Face Dataset (SMFD), and the Labeled Faces in the Wild (LFW). The SVM classifier achieved 99.64% testing accuracy in RMFD. In SMFD, it achieved 99.49%, while in LFW, it achieved 100% testing accuracy.

Keywords-twitter, python; SQLite database; insert (key words)

I. INTRODUCTION

The trend of wearing face masks in public is rising due to the COVID- 19 coronavirus epidemic all over the world. Before Covid-19, People used to wear masks to protect their health from air pollution. While other people are self-conscious about their looks, they hide their emotions from the public by hiding their faces. Scientists proofed that wearing face masks works on impeding COVID-19 transmission [1]. COVID-19 (known as coronavirus) is the latest epidemic virus that hit the human health in the last century [2]. In 2020, the rapid spreading of COVID-19 has forced the World Health Organization to declare COVID- 19 as a global pandemic. According to [3], more than five million cases

were infected by COVID-19 in less than 6 months across 188 countries. The virus spreads through close contact and in crowded and overcrowded areas. The coronavirus epidemic has given rise to an extraordinary degre of worldwide scientific cooperation. Artificial Intelligence (AI) based on Machine learning and Deep Learning can help to fight Covid-19 in many ways. Machine learning allows researchers and clinicians evaluate vast quantities of data to forecast the distribution of COVID-19, to serve as an early warning mechanism for potential pandemics, and to classify vulnerable populations. The provision of healthcare needs funding for emerging technology such as artificial intelligence, IoT, big data and machine learning to tackle and predict new diseases. In order to better understand infection rates and to trace and quickly detect infections, the AI 's power is being exploited to address the Covid-19 pandemic [4] such as the detection of COVID-19 in medical chest X-rays [5]. Policymakers are facing

a lot of challenges and risks in facing the spreading and transmission of COVID-19 [6]. People are forced by laws to wear face masks in public in many countries. These rules and laws were developed as an action to the exponential growth in cases and

deaths in many areas. However, the process of monitoring large groups of people is becoming more difficult. The monitoring process involves the detection of anyone who is not wearing a face mask. In France, to guarantee that riders wear face masks, new AI software tools are integrated in the Paris Metro system's surveillance cameras [7]. The French startup DatakaLab [8], which developed the software, reports that the goal is not to recognize or arrest people who do not wear masks but toproduce anonymous statistical data that can help the authorities predict

potential outbreaks of COVID-19. In this paper, we introduce a mask face detection model that is based on deep transfer learning and classical machine learning classifiers. The proposed model can be integrated with surveillance cameras to impede the COVID-19 transmission by allowing the detection of people who are not wearing face masks. The model is integration between deep transfer

learning and classical machine learning algorithms. We have used deep transfer leering for feature extractions and combined it with three classical machine learning algorithms. We introduced a comparison between them to find the most suitable algorithm that achieved the highest accuracy and consumed the least time in the process of training and detection. The novelty of this research is using a proposed feature extraction model have an end-to-end structure without traditional techniques with three classifiers machine learning algorithms for mask face detection. The organization for the rest of the paper is as follows. Section 2 reviews previous related works. Section 3 describes the characteristics of the

dataset. Section 4 illustrates the proposed model in detail. Section 5 reports and analyses the experimental results, and Section 6 presents the conclusions and possibilities of future work.

REQUIREMENT ANALYSIS

The project involved analyzing the design of few applications so as to make the application more users friendly. To do so, it was really important to keep the navigations from one screen to the other well ordered and at the same time reducing the amount of typing the user needs to do. In order to make the application more accessible, the browser version had to be chosen so that it is compatible with most of the Browsers.

REQUIREMENT SPECIFICATION

Functional Requirements

- Graphical User interface with the User.

Software Requirements

For developing the application the following are the Software Requirements:

- Python
- Django

Operating Systems supported

1. Windows 10 64 bit OS

Technologies and Languages used to Develop

- Python

Debugger and Emulator

- Any Browser (Particularly Chrome)

Hardware Requirements

For developing the application, the following are the Hardware Requirements:

- Processor: Intel i9
- RAM: 32 GB
- Space on Hard Disk: minimum 1 TB

INPUT AND OUTPUT DESIGN

INPUT DESIGN

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

- What data should be given as input?
- How the data should be arranged or coded?
- The dialog to guide the operating personnel in providing input.
- Methods for preparing input validations and steps to follow when error occur.

OBJECTIVES

1.Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.

2. It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.

3.When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow.

OUTPUT DESIGN

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.

2.Select methods for presenting information.

3.Create document, report, or other formats that contain information produced by the system.

The output form of an information system should accomplish one or more of the following objectives.

- Convey information about past activities, current status or projections of the
- Future.
- Signal important events, opportunities, problems, or warnings.
- Trigger an action.
- Confirm an action.

IMPLEMENTATION

MODULES

- User
- Admin
- Deep Learning
- Machine Learning

MODULES DESCRIPTION

User:

The User can register the first. While registering he required a valid user email and mobile for further communications. Once the user register then admin can activate the user. Once admin activated the user then user can login into our system. User can upload the dataset based on our dataset column matched. For algorithm execution data must be in float format. Here we too. Three face masked datasets have been selected for investigation. The Three datasets are the Real-World Masked Face Dataset (RMFD), the Simulated Masked Face Dataset (SMFD), and the Labeled Faces in the Wild (LFW). The SVM classifier achieved 99.64% testing accuracy in RMFD. In SMFD, it achieved 99.49%, while in LFW, it achieved 100% testing accuracy.User can also add the new data for existing dataset based on our Django application. User can click the Classification in the web page so that the data calculated Accuracy and F1-Score based on the algorithms. User can click Prediction in the web page

so that user can write the review after predict the review. That will display results depends upon review like positive, negative or neutral.

ADMIN

Admin can login with his login details. Admin can activate the registered users. Once he activate then only the user can login into our system. Admin can view the overall data in the browser. Admin can click the Results in the web page so calculated Accuracy and F1-Score based on the algorithms is displayed. All algorithms execution complete then admin can see the overall accuracy in web page.

DEEP LEARNING

Based on the split criterion, the cleansed data is split into 60% training and 40% test, then the dataset is subjected to four machine learning classifiers such as Support Vector Machine (SVM), ensemble algorithm. Artificial Intelligence (AI) based on Machine learning and Deep Learning can help to fight Covid-19 in many ways. Machine learning allows researchers and clinicians evaluate vast quantities of data to forecast the distribution of COVID-19, to serve as an early warning mechanism for potential pandemics, and to classify vulnerable populations. the accuracy and F1-Score of the classifiers was calculated and displayed in my results. The classifier which bags up the highest accuracy could be determined as the best classifier.

MACHINE LEARNING

Based on the split criterion, the cleansed data is split into 60% training and 40% test, then the dataset is subjected to four machine learning classifiers such as Support Vector Machine (SVM), ensemble algorithm. Artificial Intelligence (AI) based on Machine learning and Deep Learning can help to fight Covid-19 in many ways. Machine learning allows researchers and clinicians evaluate vast quantities of data to forecast the distribution of COVID-19, to serve as an early warning mechanism for potential pandemics, and to classify vulnerable populations. the accuracy and F1-Score of the classifiers was calculated and displayed in my results. The classifier which bags up the highest accuracy could be determined as the best classifier.

SYSTEM TEST

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

TYPES OF TESTS

Unit testing

 Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit

before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

Functional test

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

System Test

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot "see" into it. The test provides inputs and responds to outputs without considering how the software works.

Unit Testing

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

II. INTEGRATION TESTING

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered components on a single platform to produce failures caused by interface defects. The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.
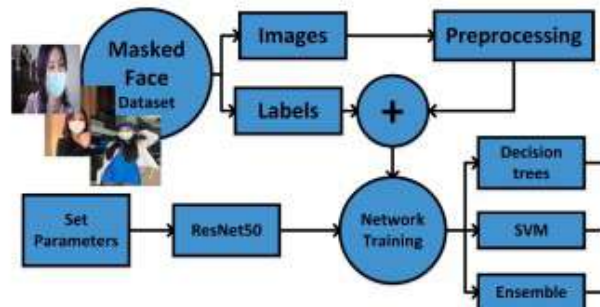
Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

SYSTEM DESIGN

SYSTEM ARCHITECTURE:

DATA FLOW DIAGRAM

1. The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.

2. The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.

3. DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.

4. DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail.
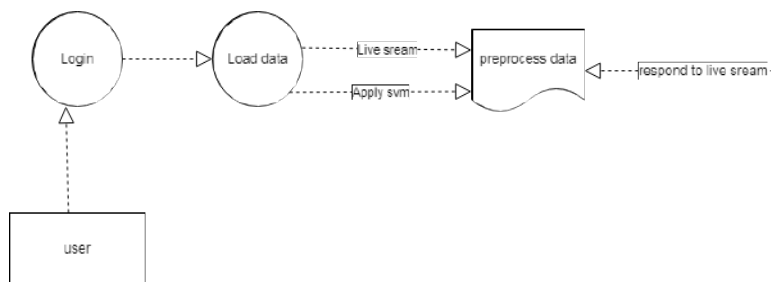
UML DIAGRAMS

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.The goal is for UML to become a common language for creating models of object oriented computer

software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems. The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

GOALS: The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modeling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

USE CASE DIAGRAM:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.
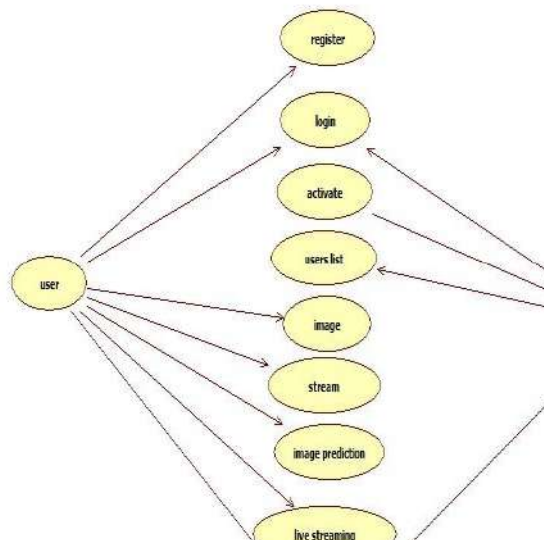
CLASS DIAGRAM:

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.
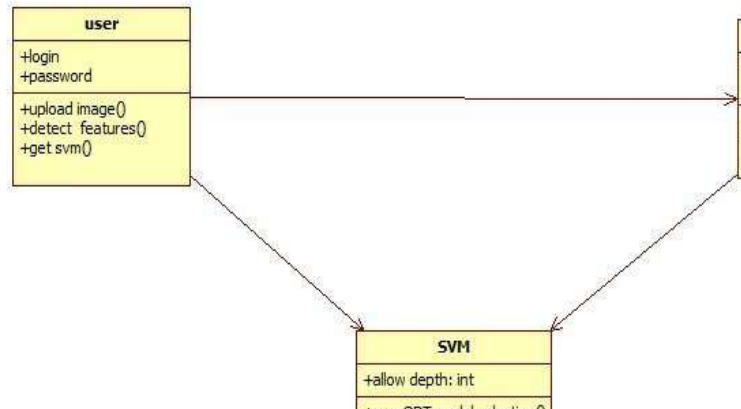
SEQUENCE DIAGRAM: A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.
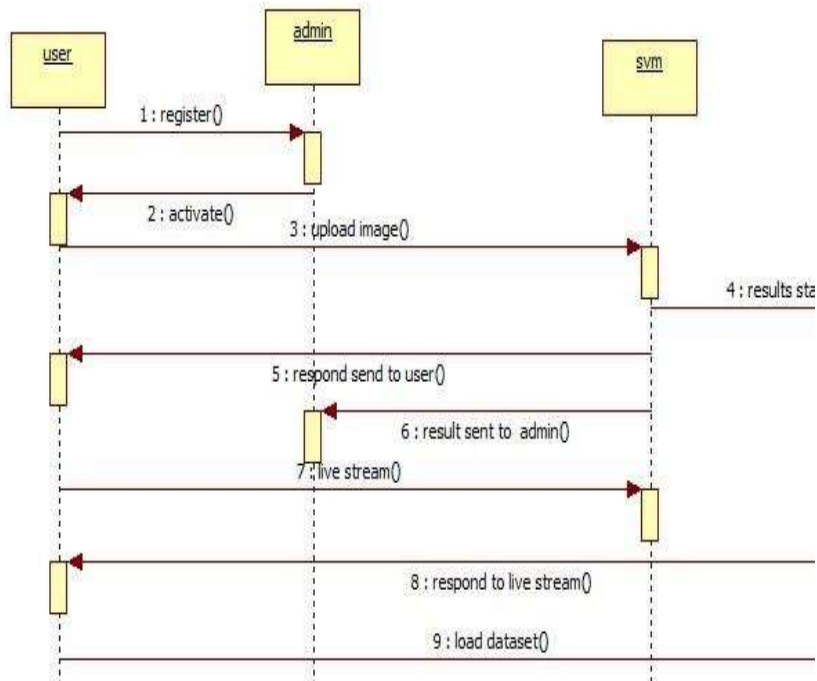
ACTIVITY DIAGRAM:

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

SAMPLE TEST CASES

S.no	Test Case	Excepted Result	Result	Remarks(IF Fails)
1.	User Register	If User registration successfully.	Pass	If already user email exist then it fails.
2.	User Login	If User name and password is correct then it will getting valid page.	Pass	Un Register Users will not logged in.
3.	User Add the Data	A new record will added to our dataset.	Pass	According to UCI repository the data must be Integer fields otherwise its failed.
4.	Data Cleaning	Data will be cleaned.	Pass	The data will be in int or float format, otherwise algorithm will not work..
5.	Tweet and Sentiment attribute plot a box	Plot based graph is generated	Pass	Target class is positive or neutral or negative label class, If not it will fail.
6.	User can add extra records for testing	User added data will be consider for testing purpose.	Pass	Data added to test data for model.
7.	Calculate f1_positve, f1_neutral, f1_negitive Scores	For our all models f1_positve, f1_neutral, f1_negitive is calculated	Pass	Data is consider for testing.
8.	Training and Testing score will be calculated	For our four models the training and testing data will be calculated	Pass	Accuracy will be consider, the failed case is data in binary format
9.	Admin login	Admin can login with his login credential. If success he get his home page	Pass	Invalid login details will not allowed here
10.	Admin can activate the register users	Admin can activate the register user id	Pass	If user id not found then it won't login.

SYSTEM STUDY

FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are,

- **ECONOMICAL FEASIBILITY**
- **TECHNICAL FEASIBILITY**
- **SOCIAL FEASIBILITY**

ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

REFERENCES

[1] S. Feng, C. Shen, N. Xia, W. Song, M. Fan, B.J. Cowling, Rational use of face masks in the COVID-19 pandemic, Lancet Respirat. Med. 8 (5) (2020) 434–436, https:// doi.org/10.1016/S2213-2600(20)30134-X.

[2] X. Liu, S. Zhang, COVID-19: Face masks and human-to-human transmission, Influenza Other Respirat. Viruses, vol. n/a, no. n/a, doi: 10.1111/irv.12740.

[3] "WHO Coronavirus Disease (COVID-19) Dashboard." https://covid19.who.int/ (accessed May 21, 2020).

[4] D.S.W. Ting, L. Carin, V. Dzau, T.Y. Wong, Digital technology and COVID-19, Nat. Med. 26 (4) (2020) 459–461, https://doi.org/10.1038/s41591-020-0824-5.

[5] M. Loey, F. Smarandache, N.E.M. Khalifa, Within the lack of chest COVID-19 X-ray dataset: a novel detection model based on GAN and deep transfer learning, Symmetry 12 (4) (2020) 651.

[6] D.M. Altmann, D.C. Douek, R.J. Boyton, What policy makers need to know about COVID-19 protective immunity, Lancet 395 (10236) (2020) 1527–1529, https:// doi.org/10.1016/S0140-6736(20)30985-5.

[7] "Paris Tests Face-Mask Recognition Software on Metro Riders," Bloomberg.com, May 07, 2020.

[8] "Datakalab | Analyse de l'image par ordinateur." https://www.datakalab.com/ (accessed Jun. 29, 2020).

[9] B. QIN and D. Li, Identifying facemask-wearing condition using image superresolution with classification network to prevent COVID-19, May 2020, doi: 10.21203/rs.3.rs-28668/v1.

[10] M.S. Ejaz, M.R. Islam, M. Sifatullah, A. Sarker, Implementation of principal component analysis on masked and non-masked face recognition, in: 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT), 2019, pp. 1–5, https://doi.org/10.1109/ ICASERT.2019.8934543.

[11] Jeong-Seon Park, You Hwa Oh, Sang Chul Ahn, and Seong-Whan Lee, Glasses removal from facial image using recursive error compensation, IEEE Trans. Pattern Anal. Mach. Intell. 27 (5) (2005) 805–811, doi: 10.1109/TPAMI.2005.103.

[12] C. Li, R. Wang, J. Li, L. Fei, Face detection based on YOLOv3, in:: Recent Trends in Intelligent Computing, Communication and Devices, Singapore, 2020, pp. 277–284, doi: 10.1007/978-981-13-9406-5_34.

[13] N. Ud Din, K. Javed, S. Bae, J. Yi, A novel GAN-based network for unmasking of masked face, IEEE Access 8 (2020) 44276–44287, https://doi.org/10.1109/ ACCESS.2020.2977386.

[14] A. Nieto-Rodríguez, M. Mucientes, V.M. Brea, System for medical mask detection in the operating room through facial attributes, Pattern Recogn. Image Anal. Cham (2015) 138–145, https://doi.org/10.1007/978-3-319-19390-8_16.

[15] M.K.J. Khan, N. Ud Din, S. Bae, J. Yi, Interactive removal of microphone object in facial images, Electronics 8 (10) (2019) , Art. no. 10, doi: 10.3390/ electronics8101115.

[16] S. A. Hussain, A.S.A.A. Balushi, A real time face emotion classification and recognition using deep learning model, J. Phys.: Conf. Ser. 1432 (2020) 012087, doi: 10.1088/1742-6596/1432/1/012087.

[17] Z. Wang, et al., Masked face recognition dataset and application, arXiv preprint arXiv:2003.09093, 2020.

[18] prajnasb, "observations," observations. https://github.com/prajnasb/observations (accessed May 21, 2020).

[19] E. Learned-Miller, G.B. Huang, A. RoyChowdhury, H. Li, G. Hua, Labeled faces in the wild: a survey, in: M. Kawulok, M.E. Celebi, B. Smolka (Eds.), Advances in Face Detection and Facial Image Analysis, Springer International Publishing, Cham, 2016, pp. 189–248.

[20] P. Khojasteh, et al., Exudate detection in fundus images using deeply-learnable features, Comput. Biol. Med. 104 (Jan. 2019) 62–69, https://doi.org/10.1016/j. compbiomed.2018.10.031.

[21] L. Wen, X. Li, L. Gao, A transfer convolutional neural network for fault diagnosis based on ResNet-50, Neural Comput. Appl. 32 (10) (2020) 6111–6124, https://doi. org/10.1007/s00521-019-04097-w.

[22] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: 2016 EEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778, https://doi.org/10.1109/CVPR.2016.90.

[23] A. Çayir, I. Yenidogan, � H. Dag, � Feature extraction based on deep learning for some traditional machine learning methods, in: 2018 3rd International Conference on Computer Science and Engineering (UBMK), Sep. 2018, pp. 494–497, doi: 10.1109/UBMK.2018.8566383.

[24] M. Jogin, Mohana, M.S. Madhulika, G.D. Divya, R.K. Meghana, S. Apoorva, Feature Extraction using Convolution Neural Networks (CNN) and Deep learning, in: 2018 3rd IEEE International Conference on Recent Trends in Electronics, Information Communication Technology (RTEICT), May 2018, pp. 2319–2323, doi: 10.1109/ RTEICT42901.2018.9012507.

[25] A. Navada, A.N. Ansari, S. Patil, B.A. Sonkamble, Overview of use of decision tree algorithms in machine learning, in: 2011 IEEE Control and System Graduate Research Colloquium, 2011, pp. 37–42, https://doi.org/10.1109/ ICSGRC.2011.5991826.

[26] P.-L. Tu, J.-Y. Chung, A new decision-tree classification algorithm for machine learning, in: Proceedings Fourth International Conference on Tools with Artificial Intelligence TAI '92, Nov. 1992, pp. 370–377, doi: 10.1109/TAI.1992.246431.

[27] Robi Polikar, Ensemble learning, in: Cha Zhang, Yunqian Ma (Eds.), Ensemble Machine Learning: Methods and Applications, Springer US, Boston, MA, 2012, pp. 1–34, https://doi.org/10.1007/978-1-4419-9326-7_1.

[28] E. Mangalova, E. Agafonov, Wind power forecasting using the k-nearest neighbors algorithm, Int. J. Forecast. 30 (2) (2014) 402–406, https://doi.org/10.1016/j. ijforecast.2013.07.008.

[29] I. Naseem, R. Togneri, M. Bennamoun, Linear regression for face recognition, IEEE Trans. Pattern Anal. Mach. Intell. 32 (11) (2010) 2106–2112, https://doi.org/ 10.1109/TPAMI.2010.128.

[30] D.G. Kleinbaum, K. Dietz, M. Gail, M. Klein, M. Klein, Logistic Regression, Springer, 2002.

[31] Y. Xiao, J. Wu, Z. Lin, X. Zhao, A deep learning-based multi-model ensemble method for cancer prediction, Comput. Methods Programs Biomed. 153 (2018) 1–9, https://doi.org/10.1016/j.cmpb.2017.09.005.

[32] C. Goutte, E. Gaussier, A Probabilistic Interpretation of Precision, Recall and FScore, with Implication for Evaluation, 2010.

[33] A. El-Sawy, H. EL-Bakry, M. Loey, CNN for handwritten arabic digits recognition based on LeNet-5 BT, in: Proceedings of the International Conference on Advanced Intelligent Systems and Informatics 2016," Cham, 2017, pp. 566–575.

[34] A. El-Sawy, M. Loey, H. EL-Bakry, Arabic Handwritten characters recognition using convolutional neural network, WSEAS Trans. Comput. Res. 5 (2017), Available: http://www.wseas.org/multimedia/journals/computerresearch/2017/a04581 8-075.php (Accessed: Apr. 01, 2020).

[35] N.E.M. Khalifa, M.H.N. Taha, A.E. Hassanien, A.A. Hemedan, Deep bacteria: robust deep learning data augmentation design for limited bacterial colony dataset, Int. J. Reason.-based Intell. Syst. (2019), https://doi.org/10.1504/ijris.2019.102610.

[36] N.E.M. Khalifa, M.H.N. Taha, A.E. Hassanien, Aquarium family fish species identification system using deep neural networks, in: Proceedings of the International Conference on Advanced Intelligent Systems and Informatics 2018, 2018, pp. 347–356.

[37] N.E.M. Khalifa, M.H.N. Taha, D. Ezzat Ali, A. Slowik, A.E. Hassanien, Artificial intelligence technique for gene expression by tumor RNA-Seq data: a novel optimized deep learning approach, IEEE Access (2020), https://doi.org/10.1109/ access.2020.2970210.