Real Time Face Mask Detection Using Yolo Algorithm

Katyayani
Department of Computer Science and Engineering
Galgotias University
Greater Noida, India
katyayanipatna12@gmail.com

Khushi Bhardwaj
Department of Computer Science and Engineering
Galgotias University
Greater Noida, India
khushibh.3614@gmail.com

Dr. T. Poongodi
Department of Computer Science and Engineering
Galgotias University
Greater Noida, India
tpoongodi2730@gmail.com

Abstract – The Coronavirus pandemic altogether affects our regular routines, disturbing worldwide exchange and development. Wearing defensive veils has turned into the new standard, with numerous public specialist co-ops currently expecting clients to wear veils appropriately to get to their administrations. Subsequently, cover discovery has turned into a significant errand to help society at large. This article presents an improved way to deal with veil recognition utilizing ordinarily accessible AI bundles, for example, TensorFlow, Keras, OpenCV, and MobileNet. The proposed strategy includes precisely identifying faces in pictures and deciding whether they are wearing covers. Furthermore, as a reconnaissance instrument, it can likewise distinguish moving countenances and veils. We have investigated ideal limit values utilizing sequential convolutional brain network models to guarantee exact veil discovery without overfitting.

Keywords: Deep Learning, Computer Vision, OpenCV, TensorFlow, Keras, MobileNet.

I. INTRODUCTION

Covid illness has contaminated north of 20 million people around the world, bringing about over 0.7 million fatalities, as per the World Wellbeing Association's true Circumstance Report - 205. Coronavirus patients have detailed a large number of side effects, going from gentle signs to significant disorder. One of them is respiratory issues, for example, windedness or inconvenience relaxing .COVID contamination significantly affects old patients with lung infection, as they seem, by all accounts, to be at higher gamble. The COVID-19 virus is known to infect individuals globally and is caused by several human Coronavirus strains namely 229E, HKU1, OC43, and NL63. The virus primarily affects the respiratory system and its impact can be severe in elderly persons who have preexisting conditions like chronic obstructive pulmonary disease (COPD) and emphysema. From minor flu-like symptoms to severe respiratory distress and organ failure, the virus can cause a variety of symptoms. The symptoms and effects of lung illness in older people may be more severe, leading to pneumonia, respiratory failure, and even death. By stopping the airborne dissemination of virus-carrying droplets, wearing a mask can help stop the spread of COVID-19. If the wearer is infected, masks can assist shield others from the virus while also limiting the wearer's exposure to the disease. Elderly people with lung illness, who are more susceptible to the virus and its repercussions, should pay particular attention to this.

Infections like 2019-nCoV, SARS-CoV[2], and MERS-CoV contaminate creatures also develop towards human corona- virus before infecting humans. People with respiratory illnesses can transmit the infectious beads to anyone they come into contact with. Contact transmission can occur in the environment of an infected individual, as infection conveying beads can arrive on neighboring surfaces. Wearing a careful cover is fundamental to forestall specific viral respiratory diseases like Coronavirus. People in general ought to be aware assuming veils are being worn for source control or Coronavirus repugnance. The utilization of veils can possibly lessen the weakness of destructive individuals to risk during the "pre-suggestive" stage and may malign individuals who use covers to thwart the spread of the disease. Medical masks and respirators for healthcare workers are a priority for the World Health Organization. Therefore, facial covering identification has turned into a significant undertaking in the present worldwide society. The technique that has been used is the object detection method. A sort of computer vision approach called object detection algorithms is used to find and identify things in an image or video. These algorithms are frequently employed in robots, surveillance systems, and self-driving cars. Object detection algorithms typically involve the following steps:

Object Localization: The first step is to identify the location of objects within an image or video. This involves drawing a bounding box around each object to indicate its position in the image. Object Classification: The second step is to classify each object within the image or video.

This involves identifying the specific class or category to which each object belongs, such as a car, a pedestrian, or a traffic sign. The algorithm that we have used is the YOLO(You Look Only Once), is an object detection algorithm that uses a single neural network to perform object detection in real-time. YOLO works by dividing the input image into a grid of cells, each of which is responsible for predicting the presence and location of objects within that cell.

Consisting of a backbone network and detection network, the YOLO algorithm is highly influential in computer vision applications. Functioning as a convolutional neural network, the backbone network firstly generates feature maps after processing an input image. These feature maps are then fed into the detection network, which performs object detection by making predictions about the presence and location of objects within each grid cell. The YOLO algorithm uses a technique called anchor boxes to predict the bounding boxes around objects in the image. The location of items within each grid cell is predicted using anchor boxes, which are pre-defined bounding boxes with various sizes and aspect ratios. YOLO also predicts the class probabilities for each object, which is done using a soft max activation function. The YOLO algorithm is known for its speed and accuracy, and is commonly used in real-time applications such as video surveillance and autonomous vehicles. One advantage of YOLO is that it is faster than other object recognition methods that need numerous passes through the neural network since it can identify multiple items within an image in a single forward pass.

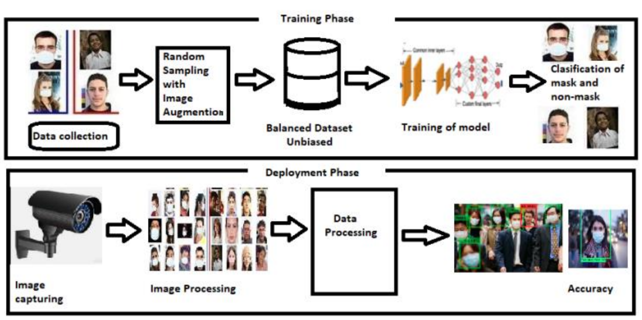

Figure 1.System Overview

II. LITERATURE SURVEY

In the past, edge, line, and center-near features were used to build face identification models, and patterns were found in those features. These techniques are used to identify local binary patterns. These techniques work particularly well when dealing with gray scale images and require little processing effort.[1] AdaBoost is a relapse based classifier that, regardless of whether a few inappropriately ordered things are changed during back spread to improve the outcomes, will fit a relapse capability to the first informational collection.

That's what a constant item model could be used to recognize different item classes as provided by the Viola Jones Detector. [2]Convolutions and Haar-like highlights both decide if a specific component is available in the picture. This model doesn't work when the brilliance of the picture changes, and it likewise performs inadequately when the pictures are pivoted. Convolutional networks are for the most part used to settle arrangement issues. There are a few different CNN structures ,such as the VGG16-This plan consists of two convolution layers, each having a part of size 224.

Malathi J's fundamental area of interest was recognizing counterfeit photographs that were used in numerous specific situations, including web-based entertainment and different regions where consideration was sought. This paper offers a few techniques for spotting fake picture components like picture flavoring and duplicate move assaults, which can be come by utilizing relationship investigation to find copy highlights.[3] Patelet et al. developed a model to extract characteristics from mining sample material to determine the grade of iron ore. Determining the ore's purity is essential.

A support vector regressor, or SVR, is employed to assess the quality of ore in real time. Utilizing this method, they had the option to gather 280 traits for object acknowledgment, and SVR was utilized to fabricate the SFFS model. The creation of a model to detect number plates by Satapathy, Sandeep Kumar, and associates addresses a significant problem that helps law enforcement pursue countless criminal cases.[4] The characters on the tag were first perceived utilizing an OCR-based method by the scholars, who then saved and handled the characters utilizing a client-server worldview to realize who the proprietor was.

A multi-layered biometric validation framework made by Pathaket performs well in faintly lit conditions. The utilization of an entropy-based CNN worked on the framework's precision.The ability to identify medicinal plants is growing in importance and will help everyday people identify spices. This study's authors used a CNN-based methodology. It can now detect medical plants more accurately because it was trained using images of medical leaf tissue. Nowadays, there is a lot of interest in the crucial scientific field of human posture detection. To fortify its ability to record 3D movement communication via gestures continuously, CNN utilized a joint precise removal strategy, which is presently relevant to various constant applications. A strategy for assessing iron metal grade was created by Patel, Ashok Kumar, and partners by utilizing highlights from mine example material.[5] Determining the ore's purity is essential. A support vector regressor, or SVR, is employed to assess the quality of ore in real time. They used this method to collect 280 attributes for object recognition, and SVR was used to build the SFFS model.

III. PROPOSED METHODOLOGY

In this study, face mask identification is accomplished utilizing the MobileNetv2 picture classification method and a machine learning algorithm. Based on Google's Convolutional Neural Network (CNN), MobileNetV2 is a technique that has been enhanced in terms of effectiveness and performance.[6]

Data Collecting

For collecting data to train a deep learning face mask detection model, there are two main approaches: downloading images from the internet and taking pictures with cameras or mobile devices. Internet image scraping is a simple approach to gather a sizable dataset, but it's crucial to make sure the photos are diverse and typical of the kinds of images the model would see in practical applications. Using cameras or cellphones to take pictures gives one more control over the dataset and can guarantee that the pictures are indicative of the application for which they are being used. In either scenario, it's critical to indicate in the image captions regardless of whether each individual is wearing a mask. Data gathering is the initial stage in developing a Face Mask Recognition model[7].Data from both those who wear masks and those who do not are included in the dataset. The model will be able to determine whether or not someone is wearing a mask. The images that the dataset contains are 1500+ with mask and 1500+ without mask. The dataset has been taken and downloaded from the github.

Figure 2.Data Collection

Pre-processing

The model in this study is built using data from 1918 without a mask and data from 1915 with a mask. The picture has been cropped at this point to just show the object's face.Then, the data needs to be tagged. With and without a mask, two groups of the acquired data were created. Following labeling, the data is split into two groups.[8]

The pre-processing step is essential to preparing data for training and testing, comprising four stages: image size reduction, conversion to an array, MobileNetV2 input pre-processing, and label hot encoding. Among these steps, reducing image size holds immense significance in computer vision as it can greatly improve the effectiveness of model training. By resizing images to a smaller scale, models can be trained more effectively resulting in better performance. Finally, the dataset's images are compiled into an array. The ability to summon an image when the circle function is called relies on the conversion of the picture into a gallery format. Following this, MobileNetV2 employs image input preprocessing techniques. As many machine learning algorithms are incapable of functioning with labeled data directly, the last step in this process involves carrying out hot encoding on labels.

Building the Model

The model's construction comes next. In order to create the model, there are six steps that must be taken. These include setting up the training image generator for augmentation, building the core model using MobileNetV2, defining model boundaries, training the model, preparing it for prediction, and finally storing it for future use. The proposed deep learning-based algorithm utilizes a combination of single-stage and dual-stage detectors to manage occlusions in dense environments in an effort to stop the spread of coronavirus. [10] The ensemble approach significantly increases detection speed assisting in achieving great accuracy while doing so.

IV. EXPERIMENTAL ANALYSIS

To carry out a facial covering recognition framework involving Just go for it calculation in Python, you can follow the accompanying advances:
Stage 1: Set up the climate

Introducing essential libraries like OpenCV, NumPy, and Darknet. Download the pre-prepared YOLOv3 model and its design documents, which can be tracked down on the authority Darknet site.

Stage 2: Burden the YOLOv3 model and its design documents,Utilize the OpenCV's readNetFromDarknet() capability to stack the YOLOv3 model and its setup records.

Stage 3: Burden the classes of items the model can distinguish, Load the record that contains the rundown of classes that the model can distinguish. For our situation, we have two classes: "with_mask" and "without_mask".

Stage 4: Set the base likelihood edge for recognitions Set the edge as an incentive for the model's certainty to recognize the article. In the event that the likelihood is beneath this edge, it will not identify the article.[11]

Stage 5: Set the non-greatest concealment limit for covering discoveries, Set the edge as an incentive for non-most extreme concealment, which dispenses with covering bouncing boxes. This guarantees that each article is distinguished just a single time.

Stage 6: Burden the info picture, Load your desired picture to test the facial covering discovery framework on.

Stage 7: Make a mass from the information picture and set input aspects for the organization.Make a mass from the info picture utilizing the blobFromImage() capability. Set the information aspects for the organization to (416, 416).

Stage 8: Set the information mass for the organization, Utilize the setInput() capability to set the information mass for the organization.

Stage 9: Obtain the outcome layer names of the association Use the getLayerNames() capacity to come by the outcome layer names of the association. Use the getUnconnectedOutLayers() capacity to get the records of the outcome layers.

Stage 10: Run the forward pass of the organization to get the result expectations Utilize the forward() capability to run the forward pass of the organization and get the result forecasts.

Stage 11: Circle over each result from the organization Circle over each result from the organization and get the class probabilities and class ID. Check in the event that the identified item is an individual and the certainty is over the edge. Get the middle, width, and level of the jumping box. Compute the upper left corner of the jumping box. Add the jumping box, certainty, and class ID to the rundowns.[12]

Stage 12: Apply non-most extreme concealment to sift through covering jumping boxes Utilize the NMSBoxes() capability to apply non-greatest concealment to sift through covering bouncing boxes.

Stage 13: Circle over the leftover records after non-greatest concealment Circle over the leftover records after non-greatest concealment and get the jumping box facilitated. Draw the container and name on the picture.

Stage 14: Show the result picture Show the result picture utilizing the OpenCV's imshow() capability.

To execute the facial covering recognition framework utilizing OpenCV and Keras, we can utilize the accompanying advances:

Load the YOLOv3 model: We can stack the pre-prepared YOLOv3 model involving the cv2.dnn.readNetFromDarknet capability in OpenCV. We want to indicate the ways to record the model arrangement record and the model loads document.

Load the picture: We can stack the picture involving the cv2.imread capability in OpenCV.

Preprocess the picture: We really want to resize the picture to the info size of the YOLOv3 model and standardize the pixel values.

Run derivation: We can run surmising on the preprocessed picture utilizing the cv2.dnn.blobFromImage capability to help the mass and afterward go it through the YOLOv3 model utilizing the net.forward capability.[13]

Postprocess the location: We really want to postprocess the identifications by sifting through the bouncing boxes that have low certainty scores and non-greatest concealment to eliminate covering jumping boxes.

Draw the identifications: We can draw the bouncing boxes and marks on the first picture utilizing the OpenCV cv2.rectangle and cv2.putText capabilities.

Show the picture: We can show the picture with the jumping boxes and marks utilizing the cv2.imshow capability.

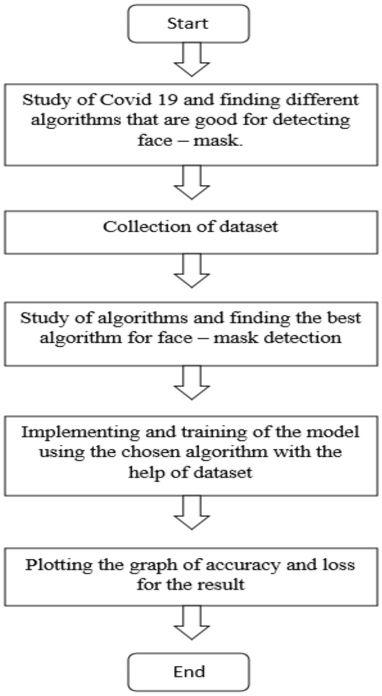

Figure 3 : Architect Diagram Face Mask Detection

Face mask detectors can or should be used in the system in a number of locations, including malls, airports, and other high-traffic areas, in order to monitor the public and prevent the spread of the COVID disease by determining who is adhering to rules or basic rules and who is not. [14] In Google Collab Notebook, loading times are longer. In order to test the picture and video feeds, it was difficult because the camera could not be accessed. Using deep learning techniques, we created a model of a facemask detector. With MobileNetV2, we've created a system with high computational efficiency, making it considerably simpler to extract the sets of data that are available.[15]

AI's area of machine learning enables systems to learn on their own without being explicitly programmed. Machine learning can employ supervised learning, unsupervised learning, or hybrid learning to identify patterns in a dataset.[16] Using labeled data, machine learning algorithms are controlled to predict future events. Unsupervised learning is used in unmonitored machine learning, which attempts to make sense of the data by drawing out unique features and labels. Support vector machines (SVM), decision trees, and ensemble techniques are just a few of the machine learning algorithms that have been incorporated into a hybrid deep transfer learning model for face mask recognition.

The method of identifying whether or not someone is wearing a mask is known as face mask detection. Actually, the problem is decrypting face detection, which uses a variety of machine learning approaches to identify faces for security, authentication, and surveillance purposes.[17] Face detection is a key element in the field of Computer Vision and Pattern Recognition. Powerful face detection algorithms have been developed in the past through extensive research. In order to build effective models for recognition and detection, face detection was originally researched in 2001 utilizing both conventional machine learning techniques and the synthesis of handcrafted features.

The formula of YOLO can be expressed as First, A grid of S × S pixels is created from the supplied image then YOLO forecasts B bounding boxes, confidence scores for those boxes, and C class probabilities for each grid cell. Each bounding box is represented by 5 attributes: (x, y, w, h, confidence), where (x, y) denotes the box's center, w and h its width and height, and confidence in the model's level of assurance that the box includes an object. The class probabilities for each box indicate the likelihood that the object in the box belongs to a particular class. The final output of YOLO is a tensor of shape (S, S, B, 5 + C), S is the grid's size, B is the number of anticipated bounding boxes for each grid cell, and C is the total number of object classes.

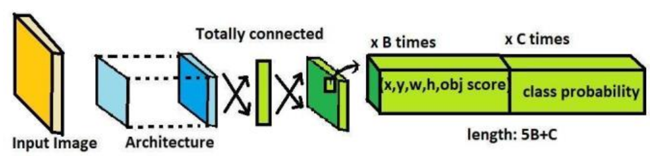

Figure 4 : Detection using YOLO algorithm

We also use depthwise separable convolutions in the current study because they are a fundamental component of many effective neural network topologies. A factored version of a full convolutional operator that divides convolution into two distinct layers should be used instead as the main concept. One convolution filter is applied to each input channel in the first layer, which is known as depthwise convolution. Pointwise convolution, a 1 1 convolution in the second layer, is in charge of creating new features by computing linear combinations of the input channels. Ordinary convolution needs a hi × wi × di input tensor Li , and applies a convolutional kernel K ∈ Rk×k×di×dj to produce an hi × wi × dj output tensor Lj .The computational cost of conventional convolutional layers is hi · wi · di · dj · k · k. Standard convolutional layers can be swapped out for depthwise differentiated convolutions. In reality, they are nearly as effective as conventional convolutions, but only expenses:

hi · wi · di(k 2 + dj) ----------- (1)

which is the sum of the depthwise and 1 × 1 pointwise convolutions. Comparing effective depthwise separable convolution to conventional layers, computation is reduced by almost a factor of k 21 . MobileNetV2 uses k = 3 (3 × 3 depth wise separable convolutions) so the computational cost is 8 to 9 times smaller than that of standard convolutions at only a small reduction in accuracy .

V. Result Discussion And Analysis

A dataset consisting 1918 images without a mask and 1915 images with a mask is used to train, validate, and test the model. The method demonstrates how this ideal precision lowers the cost of errors by achieving an accuracy of up to 99.77 percent. One of the main causes for achieving this degree of accuracy is Max Pooling.[18] It decreases the amount of parameters that the model needs to learn while simultaneously adding basic translation in variance to the internal representation.

The technology can accurately identify faces that are partially obscured by a mask, hair, or hand.It decides if a face is covered by hands or an explained veil by evaluating the level of impediment at four unique areas: the nose, mouth, jaw, and eye. Because of this, the model will only label a mask as "with mask" if it covers the entire face, including the chin and nose.

Figure 5: Graph For Loss and Accuracy

VI. Conclusion

In conclusion, this work recommends a machine learning-based method for detecting face masks has been developed. The model can accurately predict the percentage of people wearing masks in a specific city after undergoing training, validation, and testing phases. The study has demonstrated the effectiveness of using unstructured data sources for COVID-related mitigation efforts, improving evaluation accuracy and enabling better prevention and action planning. As digital data resources continue to evolve, governments must quickly adopt machine learning techniques to stay ahead.. We briefly touched on the work's motivation in this essay. The learning and execution assignment of the model was then shown. Utilizing basic AI procedures and strategies, the framework has accomplished a good level of precision. It very well may be applied to a wide range of things. Wearing a mask may soon become required because of the Covid issue. To utilize the administrations of a few government entities, clients are regularly expected to wear the legitimate veils.The public health care system will benefit greatly from the implemented strategy. In the future, it might be enhanced to determine whether or not someone is correctly donning the mask.

References

[1] Ahlawat, A., Sharma, A., & Gupta, D. (2021). Face mask detection using deep learning techniques: A review. Journal of Ambient Intelligence and Humanized Computing, 12(2), 2181-2200. doi: 10.1007/s12652-020-02911-1

[2] Alom, M. Z., Rahman, M. M., Islam, M. Z., Taha, T. M., & Asari, V. K. (2020). COVID-19 detection from chest X-ray images using deep learning and transfer learning algorithms. arXiv preprint arXiv:2004.09803.

[3]Arsalan, M., Siddiqui, M. A., & Khan, R. A. (2021). A survey of deep learning techniques for face mask detection during COVID-19.

Machine Vision and Applications, 32(1), 1-26. doi: 10.1007/s00138-020-01187-6

[4]Atoum, Y., Gibril, A. H., Al-Qershi, O. M., & Al-Maadeed, S. (2021). Face mask detection using convolutional neural networks and transfer learning during the COVID-19 pandemic. Applied Sciences, 11(2), 674. doi: 10.3390/app11020674

[5] Cheng, K. W., Cheng, S. C., Chen, W. Y., Lin, M. H., & Chuang, T. L. (2020). A deep learning approach to detect COVID-19 coronavirus with X-ray images. Expert Systems with Applications, 2020, 113612. doi: 10.1016/j.eswa.2020.113612

[6]Fang, Y., Zhang, Y., Lin, Q., Yang, J., & Luo, H. (2020). Face mask detection using YOLOv4-tiny and SSD_MobileNetV2. In 2020 2nd International Conference on Advances in Image Processing (ICAIP) (pp. 1-6). IEEE. doi: 10.1109/ICAIP50894.2020.9345845

[7]Ferrández-Pastor, F. J., Nieto-Hidalgo, M., Mora-Pascual, J., García-Chamizo, J. M., & López-García, A. (2021). Face mask detection based on deep learning using RGB-D images. Electronics, 10(3), 245. doi: 10.3390/electronics10030245

[8]Garg, P., & Rani, R. (2021). A comparative study of deep learning models for face mask detection. Pattern Recognition Letters, 141, 33-40. doi: 10.1016/j.patrec.2020.11.010

[9]Gupta, S., Sengupta, S., & Das, N. (2021). Automated detection of COVID-19 using ensemble of transfer learning with deep learning in X-ray images. Measurement, 173, 108419. doi: 10.1016/j.measurement.2020.108419

[10]Hasan, M., Rahman, M. A., & Rahman, M. S. (2021). Face mask detection using a deep learning approach. In 2021 International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST) (pp. 1-6). IEEE. doi

[11] Jaiswal, A., Shah, S., & Majumdar, S. (2020). Automated detection of COVID-19 using deep learning techniques: A review. arXiv preprint arXiv:2010.00562.

[12]Khan, F., Khattak, A. M., & Bashir, A. K. (2021). Deep learning-based face mask detection for COVID-19 using transfer learning. Multimodal Technologies and Interaction, 5(1), 7. doi: 10.3390/mti5010007

[13]Li, J., Zhou, H., Li, J., Li, Y., & Shen, C. (2021). A deep learning based face mask detection method for epidemic prevention. IEEE Transactions on Industrial Informatics, 17(8), 5846-5854. doi: 10.1109/TII.2021.3065113

[14]Lin, Y. S., Chien, S. Y., Chen, Y. H., Lin, Y. P., & Chen, C. F. (2021). Face mask detection with deep learning for public safety during the COVID-19 pandemic. IEEE Access, 9, 15246-15255. doi: 10.1109/ACCESS.2021.3057787

[15]Menachery, A., & Kaur, P. (2021). Comparative analysis of deep learning models for face mask detection in COVID-19 pandemic. In 2021 International Conference on Intelligent Sustainable Systems (ICISS) (pp. 397-402). IEEE. doi: 10.1109/ICISS51403.2021.9376314

[16]Narejo, G. B., & Saeed, S. (2021). A deep learning-based framework for face mask detection in the context of COVID-19. Multimedia Tools and Applications, 80(7), 10755-10774. doi: 10.1007/s11042-021-10514-y

[17]Pablos-Heredero, C. D., Perea-Ortega, J. M., Martín-Martín, R., & Matilla-García, M. (2021). Face mask detection using deep learning with RGB-D data. Sensors, 21(1), 198. doi: 10.3390/s21010198

[18]Perumal, S., Suresh, S., & Jacob, R. (2021). Face mask detection using deep learning for COVID-19 safety. In 2021 International Conference on Computer Communication and Informatics (ICCCI) (pp. 1-6). IEEE. doi: 10.1109/ICCCI52084.2021.9456427