Deepfake technology is a controversial technology with many wide reaching issues impacting society. Much research has been devoted to developing detection methods to reduce the potential negative impact of deepfakes. Application of neural networks and deep learning is one approach. Facebook and Instagram announced a new policy in January 2020 banning AI-manipulated ?deepfake? videos that are likely to mislead viewers. This is the focus of this paper. The results show high accuracy over all datasets with an accuracy between 91-98% depending on the deepfake technology applied. The fast evolution of deepfakes has made both the academic ?eld and technology industry put considerable focus on automated detection of deepfake videos. In this paper we consider different deep learning solutions to automatically classify and hence detect deep fake videos. We utilise FaceForen. Sics++ was used to train two neural networks using pre-processed images. Each network produces four models, eachresponding to one of four different mainstream deepfake software platforms. The result of the evaluations demonstrates a high degree of accuracy in distinguishing real and fake videos. The term deepfakes was coined from the merger of "deep learning" and "fake" It refers to the use of state-of-the-art computer vision methods and deep learning techniques to generate fake videos. Fake videos generated usingDeepfakes consist of two main categories: face-swapping and face-reenactment. Faceswap-GAN is a popular face swap method. The method is an optimized version of the original-deepfakes approach. Fast Face-swap takes A's face gesture and expression as content and B's style as style. Face-reenactment is the transferral of the expression and. pose of a source character to a character in the target video, .while the identity of the target character remains the same. Using Dlib and OpenCV, it.detects the face in the source image with the face detector, then. converts the key marker points on the face into the target. face ima. Neural Textures [9] proposes a pseudo-video generation method based on neural textures. The method uses raw video data to learn the texture of the target and incorporates information on photometric reconstruction loss. Deepfake videos or images could be distinguished from their authentic counterparts. Gardiner et al proposed a method for extracting facial regions from an image and detecting iris pixels to determine eye colour. Convolutional Neural Networks (CNNs) can detect deepfake content via image analysis features. Do et al [15] implemented a CNN with a

?ne-tuning method and achieved an accuracy above 70% using three different image datasets. FaceForensics++ provides videos that are generated from a comprehensive range of popular deepfake video generation methods. Videos contain a single unobstructed frontal face that can be easily tracked. Different video quality options are available depending on time and bandwidth constraints. The ultimate goal of this work was to identify whether a video was real, or whether it had been generated using deepfake technology. All h264 videos were downloaded to the University of Melbourne High Performance Computing (SPARTAN) system. The data was split into two parts, 80% for training the models and 20% for subsequent testing and validation. The pre-processing module needs to take into account the impact that other factors in the video may have on model training. Each frame in a video does not contain just a face. Body parts of the person and the background area of the image comprise most of the video frame. The face area in the image is the focus, and the pre- processing module must capture the face in this area as model input. The OpenCV Python package was used for this purpose. The videos selected for this work were all at a frame rate of 30 frames per second. The cascade classi?er provided by OpenCV was used. There are many deep learning models and frameworks that are now available. Xception and MobileNet were chosen as the models for the experiments in this paper for the following key reasons. The image size required for the Xception model is 299*299 while the image size for the MobileNet model is 224*224. FaceForensics provides a test environment for researchers to test their trained models. The performance of models trained by different teams are displayed in Figure 6. Among these methods, Xception shown a relatively good performance over four different datasets. Xception applies a modi?ed version of Depthwise Separable Convolution with a few key differences. Depthwise convolution is the channel-wise n*n spatial convolution while Pointwise convolution is 1*1 convolution to change the dimension. Xception does not use non-linearities in its implementation. The Xception model is lightweight,stable and time-ef?cient to implement. The focus can be on valid feature extraction and detecting deepfake videos. The Keras framework has made using Xception straightforward. MobileNets are ef?cient neural network architectures. They are suited for mobile andembedded vision-based applications that lack computational power. This architecture is based on a depthwise separable convolution. MobileNets has 28 layers in total. Each input channel has a ?lter. The model structure of MobileNets places all the computations into dense 1*1 convolutions. This is similar to Inception V3 [20] MobileNets does not tackle images with sides of heads for example. It also reduces image distortion by limiting the size of images that are allowed. SPARTAN is a high-performance HPC system operated by The University of Melbourne. The training data set and test data set were split by the video ID instead of by the picture ID. Some required packages were not available on Spartan, so a targeted virtual environment was created. When pre-processing videos, the path of output was in the form ?pictures/fake/Deepfakes/00132-00121.png? This implies that this picture was the 121st picture from video 00132. All video IDs were put into a list. This list was split into two parts with 80%used for training and 20% used for testing. The adaptive moment estimation (adam) was employed as the optimizer. A loss function based on entropy and the batch size was used. Labels were used for fake pictures (1) and for non-fake pictures (0) 20% of the videos from the dataset were tagged and reserved for the evaluation. The model?s accuracy increased gradually from epoch 1 to epoch 8, then eventually converged approaching eopch 10. In addition, the multi-processing option was set to True. The results of the experiments are shown in Figures 10-14. uracy were computed based on the following equations. The results show a high degree of accuracy of classi?cation. Both methods had relatively low detection rates on fake video datasets. MobileNet had a higher detection rate for the FaceSwap-based deepfake videos compared to Xception (by approx. 1%). A manipulated video of former president of the US, Mr Obama, was used to test the functionality of the models. The video was successfully classi?ed as fake. According to the results presented in Table 2, the video was classi ?ing as fake by a combined voting mechanism. Voting mechanism can only identify fake videos generated by the given methods. Video generated by StyleGAN2 method was used to exemplify the restrictions of the voting mechanism. Since no fake images were generated or used in the video above, it is expected that the four models would all identify the video as real. The four mainstream deepfakes methods had high detection accuracy. This was based on the premise that the corresponding detection model was used for each type of fake video. If the correspondence between the model and the testing data was not followed, the accuracy of detection is massively impacted. Existing deepfake video detection methods are designed to build video authenticity detection systems suitable for different kinds of deepfake videos. We chose to train four dichotomous classi?ers based on four dominant deepfakes methods. This detection method was able to detect all four mainstream deepfake methods with a high degree of accuracy. A voting mechanism is needed for video prediction. The voting mechanism uses the outputs of all four trained models. If any of the four model outputs are classi?ed as fake, then the video is considered fake. The Face2Face model sets a threshold such that over 50% of the images are predicted as fake. In principle, every model might have its own threshold for predicting whether a video is fake or not. For every model, the number of pictures classi?ed as fake from a real or fake video should be separated by the recommended threshold based on experimental validity. We set the threshold to 50% for all four models. In real life, there are many highly realistic but fake videos and other videos that are completely unrealistic and easy-to-detect. We do not want the threshold to be too strict because each video may have more or less pictures that are detected/undetected. The models have a similar detection accuracy for fake and real videos. A voting mechanism was implemented to utilize the four models together to detect all four types of videos generated by mainstream fake video generating methods. The classi?cation described here is based on isolated pictures obtained from videos. No inter-frame pattern correlations were considered. We did not consider video-oriented detection techniques. Authorized licensed use limited to: Central Michigan University. Downloaded on May 14,2021 at 16:58:14 UTC from IEEE Xplore. Restrictions apply. The code is available on GitHub at: http://www.gist.com/ondyari/FaceForensics/Faceforensics. /github.com/.com/shaoanlu/ faceswap-GAN. https://www.reddit.com/r/deepfakes/ ???'&22&?&22?: ??#? ?'?;. ?#': '?. ?#?: ??;. '#?: ??, '#' : ?, ?, ? #? : ?.?,. ?:. ?:. '?;. ?:. ?;:. The study was published in the open-source journal, "Theoretical Computer Science" It is based on a model of a computer program called the Pyramidean Programming Language (PPL) Authorized licensed use limited to: Central Michigan University. Restrictions apply. Downloaded on May 14,2021 at 16:58:14 UTC from IEEE Xplore.