Identification of Individuals with Down Syndrome Using Pre-Trained Models

Merve Çaylı[1*], Yakup Kutlu[1], Celaleddin Yeroğlu[1]
[1]Department of Computer Engineering, Iskenderun Technical University, Turkey
ORCID: 0009-0009-1689-118X, 0000-0002-9853-2878, 0000-0002-6106-2374
Emails: caylimerve1905@gmail.com, yakup.kutlu@iste.edu.tr , celaleddin.yeroglu@iste.edu.tr
*Corresponding author.

***Abstract*— Down syndrome is a genetic disorder and is caused by an extra copy of the 21st chromosome. This extra genetic material causes the physical and developmental characteristics associated with Down syndrome. It is known that the quality of life of individuals can be increased with developing robotic technology. This situation can be used to customize for individuals with Down syndrome, to enable them to adapt to social life more easily and to reveal their potential. The aim of the paper is to detect individuals with Down syndrome and to enable approaching these individuals differently. The outputs of this paper can be used for some devices or equipment such as humanoid robots produced for people with down syndrome. Therefore, image processing models have been developed with some methods. We tried to determine which method achieved the highest success rate of the models we developed using pre-trained models. We achieved very high success rates of around 90%.**

***Keywords*— Down syndrome, down syndrome in children, pre-trained models, deep learning**

I. INTRODUCTION

Down syndrome (DS) is the most common genomic and intellectual disability syndrome resulting from trisomy of Homo sapiens chromosome 21 (HSA21). The DS phenotype is characterized by symptoms that affect many body systems, especially the musculoskeletal, nervous and cardiovascular systems. Problems such as short stature, decreased muscle tone, cerebellar hypoplasia and congenital heart diseases are frequently observed. They are also more prone to additional health problems, such as early-onset Alzheimer's disease and hematological disorders [1]. Individuals with Down syndrome can go to school, work, and have relationships. With proper training, support and compliance, they can realize their potential. There are special education centers for individuals with Down syndrome. Here, their development is supported by receiving one-on-one training.

With developing technology, robots have begun to occupy an important place in health, education, production and many other fields. The combination of artificial intelligence, sensors, and advanced motor technologies has significantly increased the abilities of robots to perform complex tasks. This rapid development, especially in the field of healthcare, has paved the way for a number of treatments and assistive technologies that were previously considered impossible. Human-centered robotic applications, together with specialized treatment methods, have a great potential to improve the quality of life of individuals [2].

The development of individuals with Down syndrome can be positively affected by robotic technologies. A human-oriented robot should be able to distinguish individuals with Down syndrome and behave differently accordingly [3]. In this way, the development of these individuals can be supported and their adaptation to society can be contributed.

It seems that there are many studies in this field. Recently, the use of deep learning technologies has come to the fore [4,5,6]. At the same time, machine learning methods have been used together with image processing techniques using machine learning models [7,8,9]. Ali et al. [4] used the VGG16 pre-trained model, one of the deep learning technologies, in their study in 2024 and achieved 99% accuracy with Logistic Regression. In their study in 2020, Bosheng and his colleagues [5] achieved a 95.87% accuracy rate using DCNN, one of the deep learning technologies. In their study in 2021, Antoni R. and his colleagues [7] achieved a 91% accuracy rate in Congo by using Support Vector Machine (SVM), one of the machine learning methods.

Looking at previous studies, it has been seen that many different methods are used to detect individuals with Down syndrome through image processing. In this study, we will evaluate how we can get the best results by using pre-trained models used for face detection of individuals with Down syndrome.

II. MATERIALS AND METHODS

A. *Dataset*

Since many models will be tested in the paper, a large and multi-featured data set was used [8]. The images used in the dataset were collected from places with permission to use them. 1500 photographs of children with Down syndrome and 1500 photographs of healthy children between the ages of 0-18 were used.

Age groups: Although the word child covers the age range of 0-18, photographs of children between the ages of 0-15 were mostly used since there is no age information in many data and it is difficult to determine the age of individuals with Down syndrome.

Glasses: Down syndrome can also cause vision problems. This is a high rate. That's why many children with Down syndrome use glasses. However, in order to ensure that this situation does not have an impact on the decision structure of the model we will train, photos of children with down syndrome with glasses, photos of children with down syndrome without glasses, photos of healthy children with glasses and photos of healthy children without glasses were tried to be balanced.

Skin color: Since skin color is not a determining factor for Down syndrome, balanced photographs of many skin colors were tried to be used in the data set.

Eye color: Since eye color is not a determining factor for Down syndrome, balanced photographs of many skin colors were tried to be used in the data set.

Hair: Since hair color and shape and condition are not a determining feature for Down syndrome, an effort was made to keep it balanced.

B. Method

In these study, pre-trained models were used. Pre-trained models are loaded and improvements have been made on the models. Then fine tuning was done and they were trained again. Since pre-trained models are trained on very large datasets, using these models directly instead of retraining the model saves a lot of time and computational effort. Thus, it can give good results even with smaller and limited data. Adding or fine-tuning just a few layers on these models can provide effective results with less effort. In this algorithm, pre-trained models are loaded and fine-tuned. In order to achieve the desired result, some layers are not used and new layers are added.

Algorithm 1 Classification using Transfer Learning

Require: Dataset paths: train_path, test_path

-Pre-trained models: InceptionV3, Xception, MobileNet, Resnet152, DenseNet,

- Hyperparameters: epochs, patience, learning_rate

Ensure: Trained model for classification

- Performance evaluation on train and test sets

```
train_path ← "train"
test_path ← "test"
submission_file ← "sample_submission.csv"
class_names ← List of directories in train_path
total_train ← Total number of images across all classes
X ← Empty array of shape (total_train, 224, 224, 3)
y ← Zero array of shape (total_train)
for each class_name in class_names do
    for each image in class_path do
        Load image
        Resize image to (224, 224)
        Append image to X
        Append class label to y
    end for
end for
(X_train, X_test, y_train, y_test) ← train_test_split(X, y, stratify=y, test_size=0.2)
data_augmentation ← Sequential layer with RandomFlip, RandomRotation,    RandomZoom
base_model ← "Pre-trained model" with input_shape=(a, b, 3), include_top=False
Freeze base_model layers (base_model.trainable ← False)
```

```
inputs ← Input layer with shape (a, b, 3)
x ← Apply data_augmentation(inputs)
x ← Apply preprocess_input(x)
x ← Pass through base_model
x ← GlobalAveragePooling2D layer
outputs ← Dense layer with 2 units and softmax activation
model ← Create Model(inputs, outputs)
optimizer ← Adam optimizer
loss ← sparse_categorical_crossentropy
metrics ← accuracy
Compile model with optimizer, loss, and metrics
best_model ← Callback to save the best weights (based on validation loss)
early_stop ← EarlyStopping with patience=10
history ← Fit model on (X_train, y_train) with validation data (X_test, y_test), callbacks=[best_model, early_stop]
Unfreeze base_model layers (base_model.trainable ← True)
Re-compile model with a lower learning rate
Re-train model using best weights
train_proba ← model.predict(X_train)
train_prediction ← Argmax of train_proba
val_proba ← model.predict(X_test)
val_prediction ← Argmax of val_proba
model.save('model.h5')
Return: Trained model and evaluation metrics
```

III. APPLICATIONS

When choosing pre-trained models used in applications, compatibility with the data set is a very important criterion. In this context, 5 models that are successful in the field of image processing were selected and 5 applications were made. Using these applications, it can be determined which model will give the most successful results with the data set. Thus, in future robotics and similar projects, a high accuracy rate can be achieved by using this model to detect individuals with Down syndrome. In this study, instead of training many models, a single model can be trained to reduce the processing cost.

Application 1: The InceptionV3 model applies design principles such as splitting large filter sizes into smaller convolutions, optimizing spatial and depth information flow through parallel structures, and improving the training process of deep networks with assisted classifiers [9]. Since InceptionV3 is a very successful model in the field of image processing, a high accuracy rate can be expected from this model. In this application, the model was loaded using Algorithm 1. The images were resized according to the model, layers were determined and fine-tuned.

Application 2: The Xception model, which stands for "Extreme Inception", is an architecture that replaces Inception modules with deeply discrete convolutions and completely separates cross-channel and spatial correlations. [10]. The Xception model uses the architecture of the Inception model used in Application 1. However, it has the ability to capture more complex features better. Since detailed features are important, this model was preferred in Application 2. Therefore, a higher success rate can be expected from this model compared to InceptionV3. The disadvantage of this model is its computational cost. It requires a powerful GPU environment.

Application 3: DenseNet improves information flow and gradient propagation by connecting each layer to all layers before it. This dense connection structure encourages feature reuse and makes the model more parameter efficient [11]. DenseNet is a model that is thought to provide advantages in terms of time efficiency. Therefore, a good accuracy rate can be expected in terms of time/efficiency.

Application 4: MobileNet is a class of efficient deep learning models suitable for use in resource-constrained environments such as mobile devices and embedded systems. MobileNet architecture significantly reduces model size and computational cost by replacing standard convolutions with depth-wise separable convolutions [12]. MobileNet is used in this application because it is a model designed for efficiency and speed. In this model, retraining the last layers for different classes increases the success rate, so it is used in Algorithm 1. Thus, a high accuracy rate can be expected.

Application 5: ResNet152 is a model developed to overcome the training difficulties of deep neural networks and is based on the residual learning framework. This framework allows these layers to learn based on their input, rather than learning every few layers directly. ResNet152 is a deep network with 152 layers and thanks to this depth, it provides high accuracy in visual recognition tasks [13]. Resnet152 has fewer layers than other models. This may affect the accuracy rate and therefore may give lower accuracy than other models. However, this model was also used in Application 5 for comparison.

IV. RESULTS

In this study, we compared the classification performance of different deep learning models on an image dataset. In the analyzes made taking into account test and validation accuracy metrics, the InceptionV3 model showed the best performance with 91% test accuracy and 92% validation accuracy. This result reveals that the generalization ability of the model is higher than other models.

Table 1 – Performance measure of models

	Test	Accuracy Score	Validation Accuracy
1	InceptionV3	0.91	0.92
2	DenseNet	0.90	0.88
3	MobileNet	0.89	0.88
4	ResNet152	0.85	0.89
5	Xception	0.79	0.74

In contrast, the DenseNet and MobileNet models came right behind InceptionV3, with test accuracies of 90% and 89%, respectively. Both models achieved similar results in terms of validation accuracy of 88%. In particular, the small difference between the testing and validation accuracies of DenseNet demonstrates the stability and reliable performance of this model.

Although the ResNet152 model has a relatively lower performance with 85% test accuracy, it shows that its generalization ability is still strong with 89% validation accuracy. However, the decrease in testing accuracy suggests that the model may need more data or hyper-parameter optimization during training.

Finally, Xception The model exhibited the lowest performance, with 79% test accuracy and 74% validation accuracy. These results show that the model is less effective than other models for this data set.

In general, the results obtained for InceptionV3 shows that the model is the most appropriate choice for classification on this data set. However, considering the low computational costs of DenseNet and MobileNet, it has been evaluated that they can be good alternatives, especially in real-time applications.

ACKNOWLEDGEMENTS

I would like to thank Assoc. Prof. Dr. Mümine Kaya Keleş for giving me ideas and adding different perspectives.

REFERENCES

[1] Antonarakis, S. E., Skotko, B. G., Rafii, M. S., Strydom, A., Pape, S. E., Bianchi, D. W., Sherman, S. L., & Reeves, R. H. (2020). Down syndrome. NatureReviews Disease Primers, 6(1). https://doi.org/10.1038/s41572-019-0143-7

[2] Giovagnoli, R., Crucitti, A., & Dodig-Crnkovic, G. (2019). Robotics and the quality of life: The case of robotics assisted surgery. *Philosophical Inquiries*, *7*, Article 2281-8618-ETS. . https://doi.org/10.4454/philinq.v7i1.240

[3] Bargagna, S., Castro, E., Cecchi, F., Cioni, G., Dario, P., Dell'Omo, M., Di Lieto, M. C., Inguaggiato, E., Martinelli, A., Pecini, C., & Sgandurra, G. (2018). Educational Robotics in Down Syndrome: A Feasibility Study. *Technology, Knowledge and Learning*, *24*(2), 315–323. https://doi.org/10.1007/s10758-018-9366-z

[4] Raza, A., Munir, K., Almutairi, M. S., & Sehar, R. (2024). Novel Transfer Learning Based Deep Features for Diagnosis of Down Syndrome in Children Using Facial Images. *IEEE Access*, *12*, 16386–16396. https://doi.org/10.1109/access.2024.3359235

[5] Qin, B., Liang, L., Wu, J., Quan, Q., Wang, Z., & Li, D. (2020). Automatic Identification of Down Syndrome

Using Facial Images with Deep Convolutional Neural Network. *Diagnostics*, *10*(7), 487. https://doi.org/10.3390/diagnostics10070487

[6] Vorravanpreecha, N., Lertboonnum, T., Rodjanadit, R., Sriplienchan, P., & Rojnueangnit, K. (2018). Studying Down syndrome recognition probabilities in Thai children with de-identified computer-aided facial analysis. *American Journal of Medical Genetics Part A*, *176*(9), 1935–1940. https://doi.org/10.1002/ajmg.a.40483

[7] Porras, A. R., Bramble, M. S., Mosema Be Amoti, K., Spencer, D., Dakande, C., Manya, H., Vashist, N., Likuba, E., Ebwel, J. M., Musasa, C., Malherbe, H., Mohammed, B., Tor-Diez, C., Ngoyi, D. M., Katumbay, D. T., Linguraru, M. G., & Vilain, E. (2021). Facial analysis technology for the detection of Down syndrome in the Democratic Republic of the Congo. *European Journal of Medical Genetics*, *64*(9), 104267. https://doi.org/10.1016/j.ejmg.2021.104267

[8] Zhao, Q., Rosenbaum, K., Okada, K., Zand, D. J., Sze, R., Summar, M., & Linguraru, M. G. (2013). Automated down syndrome detection using facial photographs. *IEEE*, Article 978-1-4577-0216-7. https://doi.org/10.1109/EMBC.2013.6610339

[9] Burçin, K., & Vasif, N. V. (2011). Down syndrome recognition using local binary patterns and statistical evaluation of the system. *Expert Systems with Applications*, *38*(7), 8690–8695. https://doi.org/10.1016/j.eswa.2011.01.076

[10] *Detection of Down Syndrome in Children*. (n.d.). Kaggle: Your Machine Learning and Data Science Community. https://www.kaggle.com/datasets/mervecayli/detection-of-down-syndrome-in-children

[11] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the Inception Architecture for Computer Vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. https://doi.org/10.1109/cvpr.2016.308

[12] Chollet, F. (2017). Xception: Deep Learning with Depthwise Separable Convolutions. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. https://doi.org/10.1109/cvpr.2017.195

[13] Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely Connected Convolutional Networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. https://doi.org/10.1109/cvpr.2017.243

[14] Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., & Adam, H. (2017). MobileNets: efficient convolutional neural networks for mobile vision applications. arXiv (Cornell University). https://doi.org/10.48550/arxiv.1704.04861

[15] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. https://doi.org/10.1109/cvpr.2016.90