A Comparison of Convolutional Neural Networks using Transfer Learning for Cannabis Seed Classification

Giovanni Acampora[1], Carme Barrot-Feixat[2], Michele Di Nunzio[2], Marianna Santoro[3] and Autilia Vitiello[1,*]

[1]Department of Physics "Ettore Pancini", University of Naples Federico II, 80126 Naples, Italy
[2]Forensic Genetics Laboratory, Legal Medicine Unit, University of Barcelona, 08007 Barcelona, Spain
[3]IRCCS Policlinico di Sant'Orsola, Azienda Ospedaliero-Universitaria di Bologna, 40138 Bologna, Italy

Abstract

Cannabis has attracted significant attention in recent years due to its medicinal and recreational properties. However, certain cannabis varieties contain high levels of tetrahydrocannabinol (THC), which can pose health risks. This underscores the need for reliable methods to detect and classify different cannabis varieties. Traditional manual classification and sorting techniques are often time-consuming and error-prone, highlighting the potential of artificial intelligence (AI) as an effective alternative. Despite its promise, the application of AI, especially deep learning, faces challenges such as limited availability of labeled data and the necessity to deploy models on mobile devices with constrained computational resources. To address these issues, this study explores the use of transfer learning for classifying cannabis seeds. Transfer learning mitigates data scarcity by relying on pre-trained models, while computational efficiency is tackled by selecting architectures optimized for mobile environments or characterized by relatively low resource demands. The experimental evaluation, involving a new dataset comprising two cannabis seed varieties, demonstrates that deep learning models employing transfer learning can achieve high classification performance, even under resource-limited conditions.

Keywords

Convolutional Neural Networks, Transfer Learning, Seed Classification, Cannabis detection

1. Introduction

Cannabis is the botanical name of a genus within the Cannabaceae, the same plant family that contains hops. The genus includes three species, *Cannabis sativa*, *Cannabis ruderalis* and *Cannabis indica* [1]. In particular, Cannabis sativa is one of the oldest cultivated crops for various purposes such as food, medicine, and fiber. It is thought to have originated in central Asia near the northwest Himalayas and has spread throughout the world. All Cannabis plants share the presence of *cannabinoids*, or more specifically *phytocannabinoids*. *Tetrahydrocannabinol* (THC) is the most well-known cannabinoid. It shows considerable medical benefits. Indeed, THC relieves symptoms of sleep disorders, anxiety, and insomnia and acts as an antidepressant. However, high doses of THC can hurt thinking, concentration, perception, and mental function, potentially leading to behavioral disorders, hallucinations, delusions, or psychosis. Because of these side effects and risks, cannabis plants and relative seeds with high level of THC are illegal in many countries [2][3]. Hence, there is a growing demand for efficient and accurate methods to detect and classify cannabis seeds [4].

Conventional manual classification and sorting procedures are time-consuming, labor-intensive, and prone to human error. In some ways, DNA-based forensic botany techniques have outperformed conventional chemical methods in the analysis of cannabis. Nevertheless, prior efforts to find and confirm genetic markers on these synthase genes for crop type identification have run into problems [5]. In response to these challenges, artificial intelligence are emerging as a promising solution. In literature, deep learning methods have been widely used for classifications of crop seeds [6]. For example, in

1st Workshop on supporting CRIme reSolution Through Artificial INtelligence (CRISTAIN), held in conjunction with CHITALY 2025, Salerno, Italy, October 6–10

*Corresponding author.
giovanni.acampora@unina.it (G. Acampora); cbarrotfeixat@ub.ed (C. Barrot-Feixat); michele.dinunzio@ub.ed (M. Di Nunzio); marianna.santoro18@libero.it (M. Santoro); autilia.vitiello@unina.it (A. Vitiello)

[7], a Convolutional Neural Network-based model is used to classify soybean seeds. More specifically, recently, deep learning is emerging for cannabis seed classification. As an example, a non-destructive assessment of hemp seed vigor using machine learning and deep learning models with hyperspectral imaging is proposed in [8]. Another example is reported in [9], where Heo et al. developed a deep learning approach to distinguish among 17 different cannabis seed varieties using RetinaNet and Faster R-CNN by achieving good performance.

Unfortunately, one major challenge in deploying deep learning models, especially for specialized tasks such as identifying cannabis seed types is the limited availability of labeled data. Deep neural networks, particularly convolutional architectures like AlexNet, GoogLeNet, or RetinaNet, typically require large datasets to generalize well. Moreover, to make practical the use of these models, deep learning methods should run on mobile devices. However, mobile and embedded devices face hardware limitations that restrict the deployment of complex deep learning models such as limited memory and storage and low processing power. The combination of a small dataset and limited computational power complicates the use of conventional deep learning approaches on mobile platforms. Even if a model is trained offline on a high-performance server, deploying it to a resource-constrained environment requires adaptation.

Starting from these considerations, in this work, a comparison of CNN architectures using transfer learning has been carried out to classify cannabis seeds. Transfer Learning is one of the most widely used pretraining techniques which requires first training a baseline model on a large dataset and then fine-tuning it on a smaller target dataset. To create a solid foundation model, transfer learning uses parameters already trained on other source data rather than explicitly training the model on a comparatively small target dataset [10]. This characteristic of transfer learning significantly reduces the need for large amounts of training data, which is typically required by deep learning methods [11]. Moreover, to better reflect real-world scenarios where CNNs are expected to run on mobile devices, this study focuses on four CNN architectures, MobileNet [12], EfficientNet [13], Xception [14], and NASNetMobile [15], which are either specifically optimized for mobile use or are characterized by relatively low computational cost. The proposed models were tested in an experimental session involving a new dataset composed of hemp and marijuana seeds and evaluated through standard performance metrics for classification tasks.

2. Methods

In our study, four different state-of-the-art pre-trained CNN architectures have been used to extract features and classify images of cannabis seeds: MobileNet [12], EfficientNet [13], Xception [14] and NasNetMobile [15].

The MobileNet family comprises multiple variants of CNNs designed to be lightweight and efficient, making them ideal for use on mobile and embedded devices with limited computational resources. It breaks down the standard convolution into two separate operations, *depth-wise convolution layers* and *point-wise convolution layers*, reducing the number of parameters and computations required while maintaining or even improving accuracy. MobileNetV2 [16] is the architecture chosen in our experiments. It keeps efficiency and offers better performance compared to the original MobileNet thanks to inverted residual blocks and linear bottleneck layer. MobileNetV2 includes a set of hyper-parameters that allow developers to balance between accuracy, speed, and resource usage depending on their specific needs. Among these, one of the most important hyper-parameter is the width multiplier, often denoted as α. This parameter controls the number of channels (or filters) in each layer of the network. By setting α to a value less than 1 (such as 0.75 or 0.5), the model becomes narrower, significantly reducing both the number of parameters and the amount of computation. This comes at the cost of some reduction in accuracy. Conversely, values greater than 1 (like 1.3 or 1.4) can be used to create larger models for higher accuracy when computational resources are not a constraint.

EfficientNet is a family of CNN architectures developed by Google AI, designed to achieve high accuracy with significantly fewer parameters and lower computational cost compared to traditional

models. The key innovation behind EfficientNet is the use of a compound scaling method, which uniformly scales the network's depth, width, and resolution using a set of fixed scaling coefficients. EfficientNetB0, part of the EfficientNet family, is the variant used in our experiments. This variant is specifically designed to achieve high performance while being computationally efficient. EfficientNet-B0 starts with a stem convolution and is followed by seven stages of Mobile Inverted Bottleneck Convolution (MBConv) blocks, each with specific kernel sizes, strides, and expansion ratios. Inside the convolutional layers and MBConv blocks, the activation function used is Swish. Instead, at output layer, softmax function is typically used.

Xception is a deep convolutional neural network architecture that takes the original Inception idea to its logical extreme by completely replacing Inception modules with depthwise separable convolutions. In detail, Xception's architecture is composed of three main flows: 1) Entry Flow composed of a few regular convolutions followed by depthwise separable convolutions; 2) Middle Flow composed of identical blocks of depthwise separable convolutions with residual connections inspired by the success in ResNet; 3) Exit Flow composed of a final set of depthwise separable convolutions and max-pooling, followed by global average pooling and fully connected layers. Its hierarchical structure aids in learning hierarchical representations and facilitates the flow of information through the network.

NasNetMobile is a convolutional neural network designed using Neural Architecture Search (NAS), i.e. an automated process that uses machine learning to discover high-performing architectures. Indeed, NAS framed the problem of finding the best CNN architecture as a Reinforcement Learning problem through its three main constituents: *search space*, *search strategy*, and *performance estimation*. The search space defines the set of possible architectural choices that can be explored. The search strategy determines how the NAS algorithm explores the search space to find the best architecture. Once a candidate architecture is selected, the evaluation strategy measures its performance.

3. Experiments and Results

This section is devoted to describing data, experimental setup and results of our comparison study.

3.1. Dataset

A total of 84 cannabis seeds, including marijuana and hemp from various geographical origins, were used in this study. Seeds were legally purchased online or from authorized cannabis shops:

- Italian hemp (N = 10), CB Weed (Le Marche);
- Czech hemp (N = 10), Cannandorra Hemp (CZ-BIO-002);
- Danish hemp (N = 10), Raab Vitalfood (DE-OKO-001);
- Danish hemp (N = 10), Reformhaus Hemp (DE-OKO-003);
- Spanish hemp (N = 10), Gramso (Comunidad Valenciana);
- Marijuana (N = 34), Royal Queen Seeds.

Despite their diverse origins, all samples were included in a binary classification task (hemp vs. marijuana).

Each seed was imaged using a fully automated inverted fluorescence microscope (Leica DMI6000b). Images were captured at 4000 × 3000 pixels (96 dpi, both horizontal and vertical resolution) and saved in .JPG format.

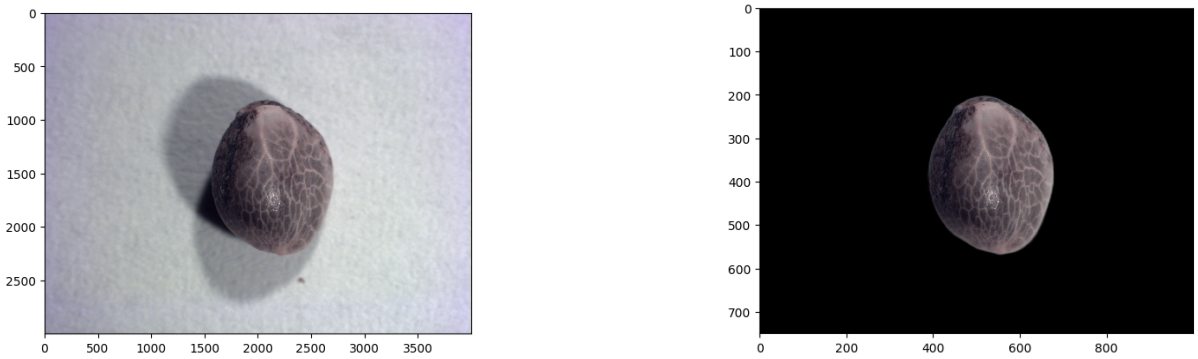
3.2. Experimental setup

Various preprocessing methods were used on our dataset to improve the quality of the input data, make the model training more efficient, and ensure model generalization. These images were pre-processed in the following sequence:

- Data labeling to categorize hemp and marijuana classes;

- Transform non-numerical labels into ordinal encoding scheme;
- Seed segmentation using Rembg [17] tool based on U2-Net to remove the background while keeping salient seed features;
- Resize operation from 4000x3000 pixels to 1000 × 750 pixels for training efficiency;
- Data normalization (pixel values scaled between 0 and 1) to make the network learning faster and converge more efficiently.

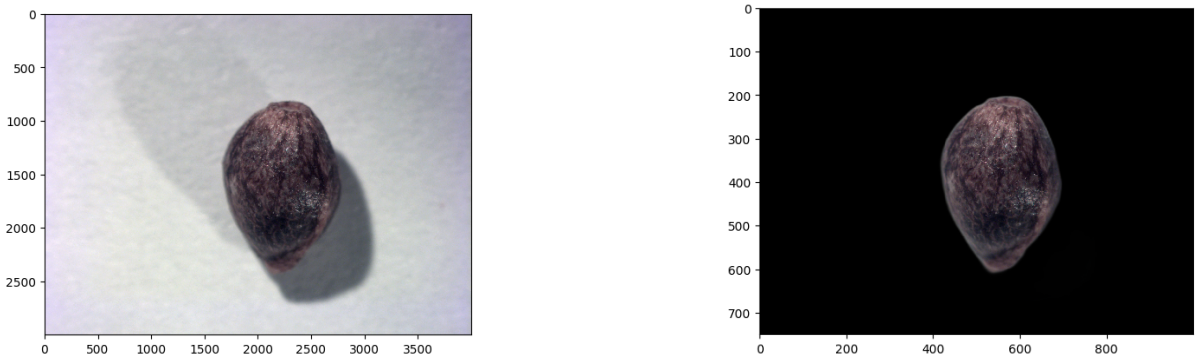
Figs. 1 and 2 show the effect of the preprocessing operations on the images, after segmentation and resizing. From the images it is possible to deduce that the transformations have not worsened the quality of the images and the salient features are still well visible and optimized for the training process.



(a) Original hemp seed

(b) Hemp seed after preprocessing

Figure 1: Preprocessing on an image related to one hemp seed.



(a) Original marijuana seed

(b) Marijuana seed after preprocessing

Figure 2: Preprocessing on an image related to one marijuana seed.

After these initial image preprocessing steps, we split the dataset in the ratio of 80:20 for training and test folders, ensuring class balancing. Validation data have been obtained from training data with a ratio of 90:10 for training and validation folders for monitoring the model's generalization performance. For feature extraction, we have used multiple pre-trained CNN architectures as mentioned before from Keras Applications. Default hyper-parameters were used. In particular, $\alpha = 1$ for MobileNetV2 allows a good trade-off between performance and computational cost.

We get pre-trained weights alongside each model. All of them were trained on the well-known ImageNet dataset [18], which consists of 1000 object category classifications. There are more than a million images in ImageNet's training set, around fifty thousand in its validation set, and one hundred thousand in its test set. Freezing the pre-trained layers allows the network to focus on learning new, task-specific features while leveraging the knowledge already encoded in the pre-trained model. Using Transfer learning-based models has the advantage of reducing model training time, reducing

generalization errors, and the need for huge datasets. In the Transfer learning scenario, the following steps were computed:

- Retrieving layers of the pre-trained model and download the pre-trained weights;
- Freezing layers to avoid weights being re-initialized;
- Adding new trainable layers on top of frozen layers that will turn old features into predictions on our new dataset;
- Training new layers on our seed dataset.

The added layers in our networks are the GlobalAveragePooling2D and Dense layers. The GlobalAveragePooling2D layer is used to extract spatial information from feature maps. It's instrumental in reducing parameters and simplifying model architectures, as it contains fewer parameters than the Flatten layer, which reduces the risk of overfitting and helps build a more efficient model. The final layer of the model is the Dense layer, The goal is to use the pre-trained model, or a part of it, to pre-process images and get essential features and pass these features to this new classifier with no need to retrain the base model. The Dense layer contains 1 neuron because there are two possible classes, hemp and marijuana, and a Sigmoid activation function to output a probability between 0 and 1.

Details of the training settings are given in Table 1. In this configuration scenario, all 4 models contained frozen and trainable parameters whose size is displayed in Table 2

Table 1
Hyperparameter training settings.

Hyperparameter	Value
Optimizer	Adam
Learning Rate	0.001
Epochs	20
Loss	BinaryCrossEntropy

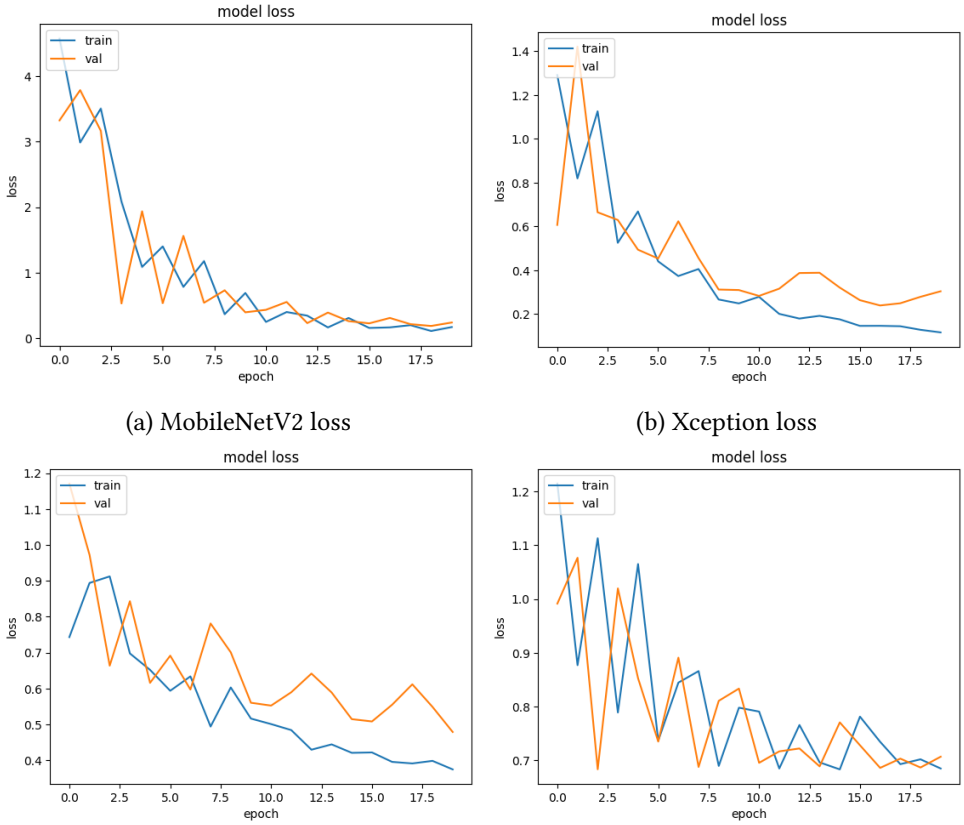
Table 2
Number of trainable and non-trainable parameters

Architecture	Frozen	Trainable
EfficientNetB0	4,049,571	656,385
MobileNetV2	2,257,984	656,385
Xception	20,861,480	1,049,601
NasNetMobile	4,269,716	541,697

Performance evaluation methods such as Accuracy, Precision, Recall, and F-score were used to evaluate models created for our binary classification task. These metrics were obtained from the confusion matrices where true positives, false positives, true negatives and false negatives are reported. It is important to note that, in a forensic context, if the positive class is defined as representing illegal seeds, the presence of false negatives in cannabis seed classification means that some illegal seeds may go undetected, potentially leading to serious operational consequences.

3.3. Results

Table 3 shows the training and test performance metrics of each compared model taken as a weighted average (taking the support of each class into account) on the acquired dataset.



(a) MobileNetV2 loss

(b) Xception loss

(c) NasNetMobile loss

(d) EfficientNetB0 loss

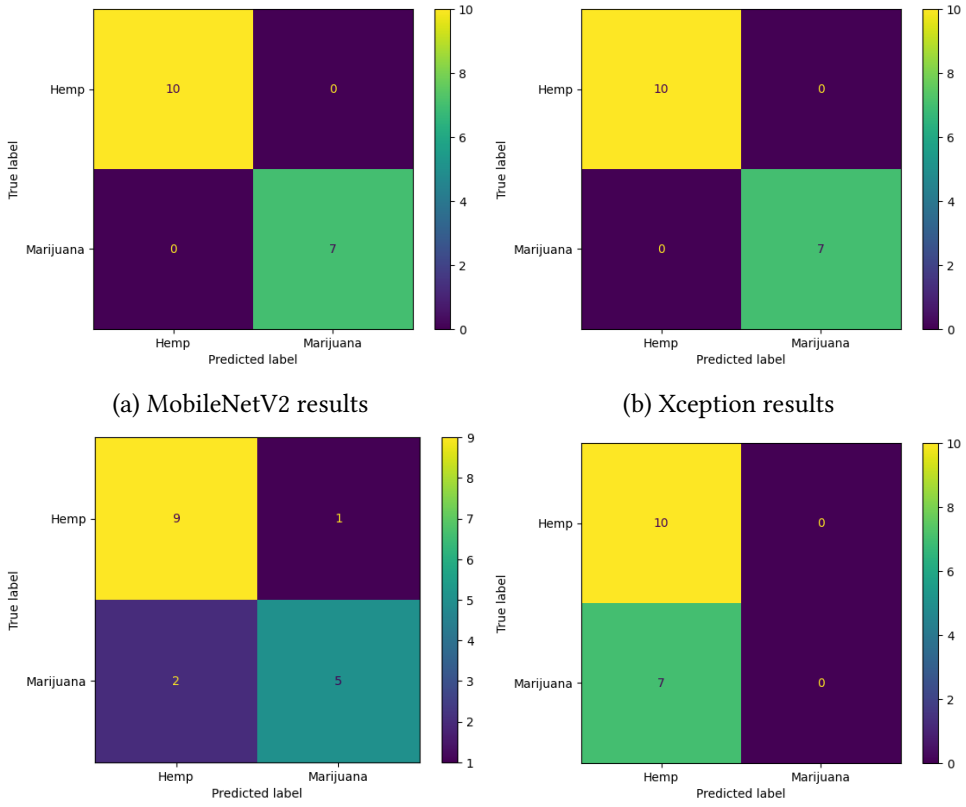
Figure 3: Training and validation loss for each proposed pre-trained architectures.

Table 3
Performance evaluation metrics comparison of each model on training and test data.

Performance metrics on training data				
Model	**Precision**	**Recall**	**Accuracy**	**F1-Score**
MobileNetV2	0.98	0.98	0.98	0.98
Xception	0.98	0.98	0.98	0.98
NasNetMobile	0.85	0.85	0.85	0.85
EfficientNetB0	0.36	0.60	0.60	0.45
Performance metrics on test data				
Model	**Precision**	**Recall**	**Accuracy**	**F1-Score**
MobileNetV2	1.00	1.00	1.00	1.00
Xception	1.00	1.00	1.00	1.00
NasNetMobile	0.82	0.82	0.82	0.82
EfficientNetB0	0.35	0.59	0.59	0.44

EfficientNetB0 produced worse outcomes on both training and testing data. Neither meaningful patterns nor effective generalization were learned by the model. Due to the small dataset size and lack of training time, the model may not have learned the relevant patterns in our dataset, which could lead to poor performance. As shown in Fig. 3, validation loss remains high after 20 epochs, suggesting the need to use more epochs and data-augmentation technique to fine-tune the weights properly.

MobileNetV2, Xception and NasNetMobile models confirmed their strength even in the presence of small data, and they are able to converge, overcoming the trade-off between the amount of data vs. model complexity. As expressed in Table 3, MobileNetV2 and Xception achieved the best metrics for marijuana/hemp classification, achieving a 100% score for each metric performance on test data. They are



(a) MobileNetV2 results

(b) Xception results

(c) NasNetMobile results

(d) EfficientNetB0 results

Figure 4: Confusion Matrices of the compared CNN architectures.

followed by NasNetMobile, with lower performance in terms of accuracy, Recall, F1-score, and Precision (82% for all metrics). The above metrics are calculated with the help of confusion matrices reported in Fig. 4 which provide a detailed breakdown of how well each classification model performs. Confusion matrices confirm the robustness of the models, with the exception of the EfficientNet architecture which, as can be seen, incorrectly classifies all marijuana samples as hemp seeds.

To conclude, considering also the number of trainable parameters in the comparison, the MobileNetV2 model emerges as the most efficient choice, as it achieves the same performance of Xception while requiring significantly fewer parameters to be trained.

4. Conclusions

As part of our study, four state-of-the-art CNN-based transfer learning models, such as MobileNetV2, Xception, EfficientNetB0 and NasNetMobile were trained on a new dataset of marijuana and hemp seeds. All four architectures are designed with efficiency in mind, specifically focusing on achieving high accuracy while keeping the model size and computational cost low. The best accuracy results were obtained with MobileNetV2 and Xception architectures, reaching 100% accuracy on test data, correctly classifying each seed class. However, when considering the number of trainable parameters, MobileNetV2 proves to be the best option, matching Xception's performance with a much smaller model size. In general, except for EfficientNet, the considered CNN architectures using transfer learning offer good performance and computational efficiency, making them a good choice for mobile applications also in the forensic context.

Acknowledgments

This study was funded by Ministero dell'Università e della Ricerca (MUR) of Italy in the context of project denoted as BLOODSTAIN in the program PRIN 2022 (grant number E53D23008040001).

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

[1] S. Schilling, R. Melzer, P. F. McCabe, Cannabis sativa, Current Biology 30 (2020) R8–R9.
[2] H. P. Devkota, Hemp (cannabis sativa l.)-taxonomy, distribution and uses, in: Revolutionizing the potential of hemp and its products in changing the global economy, Springer, 2022, pp. 1–10.
[3] S. Lee, Y. Lee, Y. Kim, H. Kim, H. Rhyu, K. Yoon, C.-D. Lee, S. Lee, Beneficial effects of cannabidiol from cannabis, Applied Biological Chemistry 67 (2024) 32.
[4] T. Islam, T. T. Sarker, K. R. Ahmed, N. Lakhssassi, Detection and classification of cannabis seeds using retinanet and faster r-cnn, Seeds 3 (2024) 456–478.
[5] Y.-C. Cheng, R. Houston, The development of two fast genotyping assays for the differentiation of hemp from marijuana, Journal of Forensic Sciences 70 (2025) 49–60.
[6] V. Kumar, P. S. S. Aydav, S. Minz, Crop seeds classification using traditional machine learning and deep learning techniques: A comprehensive survey, SN Computer Science 5 (2024) 1031.
[7] Z. Huang, R. Wang, Y. Cao, S. Zheng, Y. Teng, F. Wang, L. Wang, J. Du, Deep learning based soybean seed classification, Computers and Electronics in Agriculture 202 (2022) 107393.
[8] D. Onwimol, P. Chakranon, K. Wonggasem, P. Wongchaisuwat, Non-destructive assessment of hemp seed vigor using machine learning and deep learning models with hyperspectral imaging, Journal of Agriculture and Food Research (2025) 101836.
[9] T. Islam, T. T. Sarker, K. R. Ahmed, N. Lakhssassi, Detection and classification of cannabis seeds using retinanet and faster r-cnn, Seeds 3 (2024) 456–478.
[10] Z. Zhao, L. Alzubaidi, J. Zhang, Y. Duan, Y. Gu, A comparison review of transfer learning and self-supervised learning: Definitions, applications, advantages and limitations, Expert Systems with Applications 242 (2024) 122807.
[11] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, C. Liu, A survey on deep transfer learning, in: Artificial Neural Networks and Machine Learning–ICANN 2018: 27th International Conference on Artificial Neural Networks, Rhodes, Greece, October 4-7, 2018, Proceedings, Part III 27, Springer, 2018, pp. 270–279.
[12] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, H. Adam, Mobilenets: Efficient convolutional neural networks for mobile vision applications, arXiv preprint arXiv:1704.04861 (2017).
[13] M. Tan, Q. Le, Efficientnet: Rethinking model scaling for convolutional neural networks, in: International conference on machine learning, PMLR, 2019, pp. 6105–6114.
[14] F. Chollet, Xception: Deep learning with depthwise separable convolutions, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 1251–1258.
[15] B. Zoph, V. Vasudevan, J. Shlens, Q. V. Le, Learning transferable architectures for scalable image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 8697–8710.
[16] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L.-C. Chen, Mobilenetv2: Inverted residuals and linear bottlenecks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 4510–4520.
[17] D. Gatis, rembg, 2025. URL: https://github.com/danielgatis/rembg.
[18] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla,

M. Bernstein, et al., Imagenet large scale visual recognition challenge, International journal of computer vision 115 (2015) 211–252.