REPORT OF SMALL SAMPLE LEARNING FOR EFFICIENT DOMAIN-SPECIFIC IMAGE BINARY CLASSIFIER

GROUP NAME: GIVE ME FIVE

Boyang Peng (z5499630)

1. Introduction

Machine learning's ability to learn from small sample datasets has become pivotal in an era of data-driven decision-making. The challenges provided by BarriJam encompass real-world binary classification problems with limited sample sizes, which push the envelope of model design and drive us toward technological innovation. In this report, our team will explore how machine learning techniques, especially on small sample datasets, can efficiently solve real-world binary classification problems.

Project Tasks:

1. Identifying whether a piece of music is of the epic genre from its spectrogram.
2. Determining the need for weed removal from images of pavements with weeds.
3. Identifying whether a photograph is AI-generated.

Each scenario utilizes datasets with very few labeled samples, testing our model design and driving technical innovation.

Main Issues:

The obvious primary issue at hand was the limited number of samples available for training, which inherently increases the risk of overfitting, where a model learns the training data too well and fails to generalize to new data. This challenge is compounded by the diverse nature of the datasets and the complexity of the tasks, which require the model to discern subtle patterns and make predictions with high confidence.

The Approaches:

To address the limitations imposed by small sample sizes, our approach was in three parts:

1. Model Selection: We experimented with a diverse range of models provided by Pytorch[1], from traditional algorithms like SVM[2] to modern

neural network architectures like MobileNetV3 and VGG16[4]. This allowed us to explore the spectrum of model capabilities and identify which were best suited for our specific tasks.

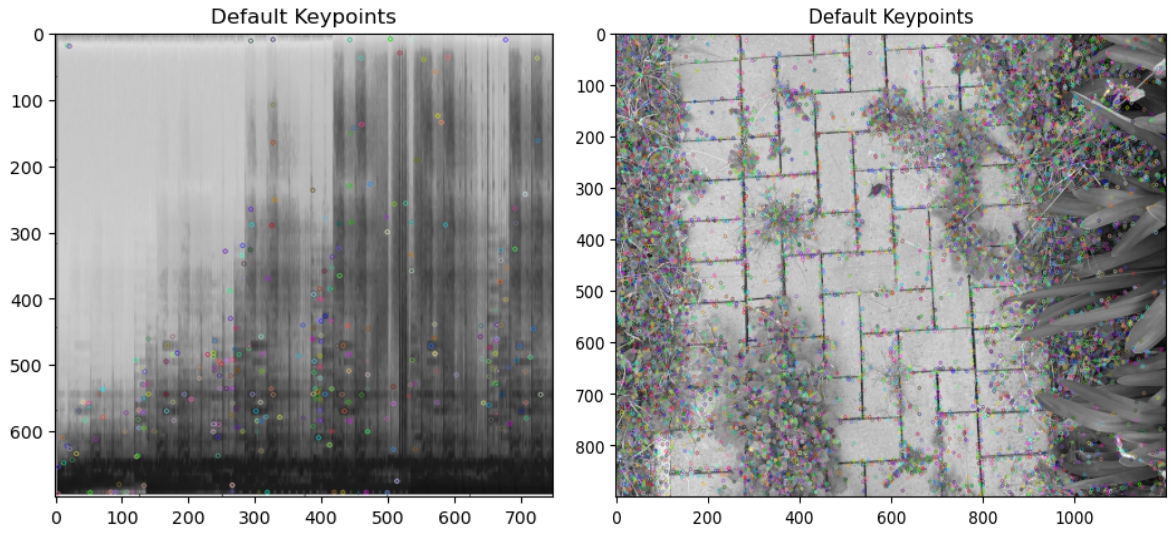
2. Data Augmentation: Recognizing the need to enhance our dataset without compromising its integrity, we employed data augmentation techniques. These included geometric transformations like rotation and, as well as photometric adjustments like ColorJitter. These techniques effectively expanded our dataset, providing the models with a richer learning experience.
3. Transfer Learning: To capitalize on the knowledge gained from large datasets, we implemented transfer learning strategies, particularly with CNN architectures. By starting with models pre-trained on extensive datasets, we adapted the learned features to our smaller datasets, thereby improving learning efficiency and boosting performance.

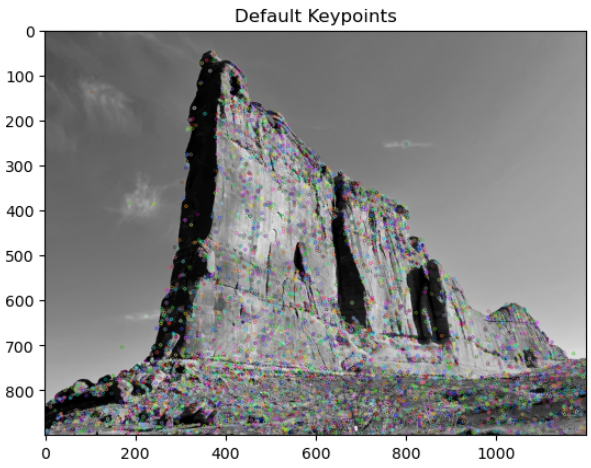
By using these models: SVM, VGG16, MobileNet V3 and simple CNN, after training and testing, MobileNet V3 was chosen as the final model due to its advantages compared by other models. Through this machine learning, we can predict accurately just by learning from small datasets, which can be utilized in many fields and make our life more convenient and efficient in the future.

2. Exploratory Data Analysis

When analyzing images, extracting keypoints is a crucial step in understanding and processing the visual information they contain. They are distinct and easily recognizable parts of an image.

The distribution and density of keypoints across the given images below provide a glimpse into the areas with high information content. This is particularly useful in focusing processing resources on regions of interest within the image.





Furthermore, by comparing the keypoints across different images, we can assess similarities and differences, which aids in tasks such as image classification or object detection in diverse conditions like our project.

3. Methods

3.1 Data pre-processing procedure

Introduction to datasets

The datasets described here are designed for binary image classification tasks, each targeting a specific challenge. The first dataset, "Epic Intro," aims to classify spectrogram images of the first 30 seconds of a song to determine whether they have an epic quality, useful for matching music with visual or narrative content in the media industry. It contains 10 images, split equally between epic and non-epic examples.

The second dataset, "Needs Respray," is utilized for a vision system in a robotic weed management application, determining from images if weeds between pavers are alive and need respraying or are dead and require no further action. This dataset includes 12 images with an equal split between live and dead weeds.

Finally, the "Is Gen AI" dataset is provided to differentiate AI-generated images from real photos taken by a camera, a critical tool for verifying authenticity in the era of generative AI. It comprises 20 images, evenly distributed between AI-generated and camera-taken photos, provided in pairs.

Data augmentation

Each of our initial three datasets contains only more than ten images. In order to enrich the diversity of samples and prevent overfitting, the images in the original dataset should be amplified first.

We first tried to synthesize new images by generating adversarial networks. In the training process of GAN, the generator and discriminator are trained against each other. The generator tries to generate pictures that deceive the discriminator, while the discriminator tries to improve its discriminating ability. However, due to the limited performance of the graphics card, only 1000 images with 128*128 resolution were generated. These images lost a lot of

feature details, which made subsequent training of the classifier less effective, and we eventually abandoned this method.

We then performed data augmentation on the original dataset. The original pictures were randomly flipped, rotated at a certain angle randomly, and the contrast, brightness and saturation were randomly adjusted. The original ten pictures in the three data sets were expanded to 800, 800 and 1400 pictures respectively, and then the size of all pictures was resized to 224. Data augmentation greatly improves the diversity of samples and the generalization ability of models.After data augmentation, we divided data set into training set and validation set in an 80:20 ratio[3].

3.2 Model introduction and selection

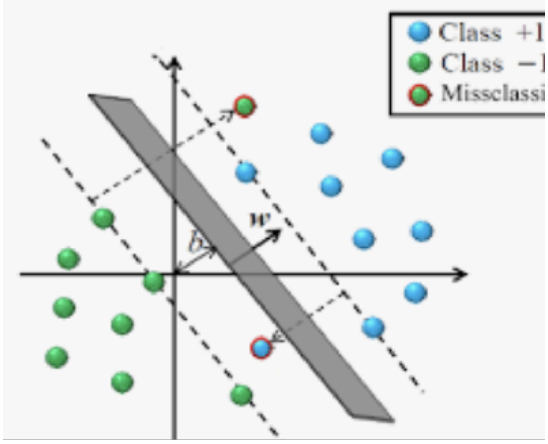


Figure 1 SVM classification diagram

SVM: A binary classification model that tries to find an optimal hyperplane that separates different classes of data points, maximizing spacing while minimizing classification errors.

Our model uses the BoVW method. Firstly, images are loaded from the training data and SIFT feature descriptors are extracted. Then KMeans is used to cluster the feature descriptors and generate visual words. The visual word histogram of each image is constructed as the feature representation of the image. Then, SVM is used to train the image features and learn how to map the feature histogram of the image to the corresponding category. Finally, the same features

are extracted and represented on the test set, and then the trained SVM model is used for prediction and evaluation.

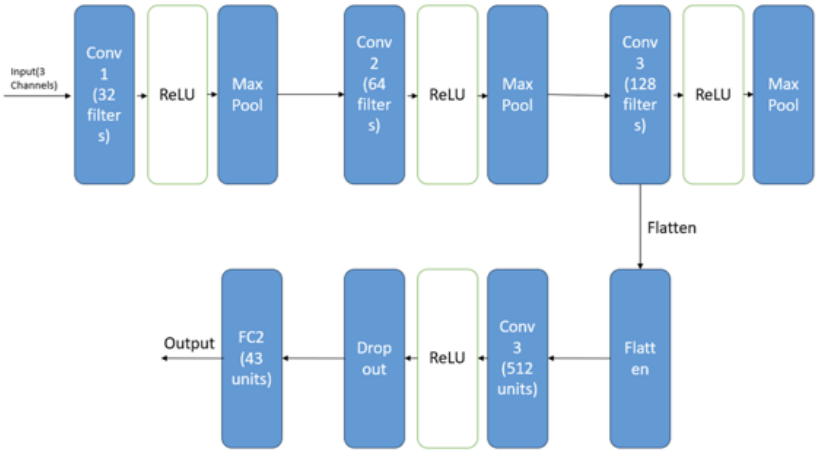


Figure 2 Structure of Simple Custom CNN

Custom CNN: Our custom Convolutional Neural Network (CNN) architecture is specifically designed for the task at hand. It incorporates three convolutional layers, which are essential for capturing spatial hierarchies in the images. Each convolutional layer is followed by a maximum pooling layer that reduces the spatial size of the representation, thus reducing the number of parameters and computation in the network. This design helps in extracting prominent features while maintaining computational efficiency. The network concludes with two fully connected layers that synthesize the data extracted by the convolutional layers to form the final output predictions.

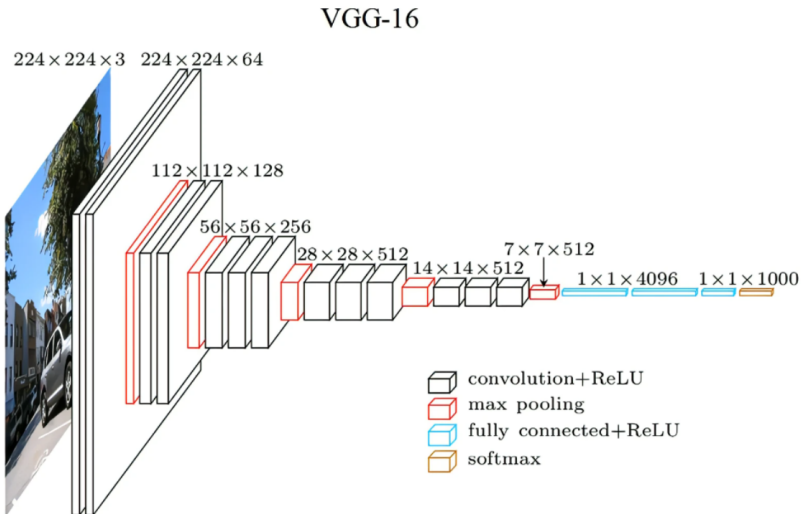


Figure 3 Structure of VGG-16

VGG16: This is a classical convolutional neural network architecture for image classification. The network uses continuous small convolutional nuclei (3x3) and pooling layers to build a deep neural network. The depth of the network reaches 16 layers, including 13 convolutional layers and 3 fully connected layers. It consists of multiple convolution layers and pooling layers stacked alternately, and finally uses the fully connected layer for classification.

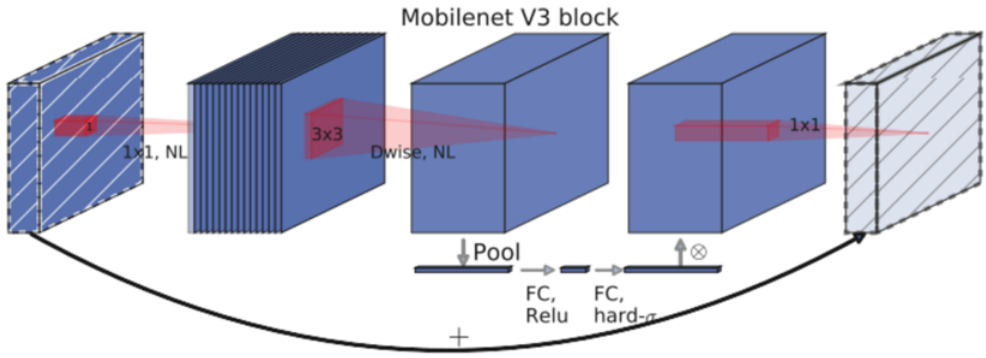


Figure 4 Structure of MobileNet V3

MobileNet V3: This model is a streamlined architecture designed for mobile and resource-constrained environments. It incorporates advanced techniques such as depth-separable convolutions, which split the convolutional process into depthwise and pointwise operations, significantly reducing computational load and model size. MobileNet V3 also utilizes a structure that varies the width of the network to optimize performance and incorporates network pruning, which eliminates redundant connections and weights for enhanced efficiency. For our specific task of binary classification, we have modified the architecture's final layer to adapt the output to a two-class system effectively.

3.3 Training process and parameter tuning

We used hyperparameter tuning in the initial model training. The learning rate, Adam optimizer, batch size, epoch size and other parameters were adjusted, but the performance of the model was still not good.

In order to further improve the model training effect, we used the pre-training weights based on torchvison library on mobile net. The pre-training method

makes the initial weights relatively reasonable, avoids the training from scratch, and improves the generalization ability of the model to new data.

3.4 Valuation metrics

Accuracy:

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}}$$

Accuracy is the ratio of the number of samples correctly classified by the model to the total number of samples, and is one of the most commonly used evaluation indicators.

F1 score:

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \qquad \text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

F1 score considers both accuracy rate and recall rate, and is the harmonic average of accuracy rate and recall rate, which can comprehensively evaluate the performance of each model.

Since the classification task has a 10-minute time limit, the training time and reasoning time of the model should also be used as an evaluation index. The performance, accuracy and real-time performance of the model should be considered comprehensively to weigh and compare.

4. Results and Discussion

In the following sections, we sequentially present the classification results based on the methods employed by our chosen models.

The first set of results pertains to the "Is Epic Intro" dataset:

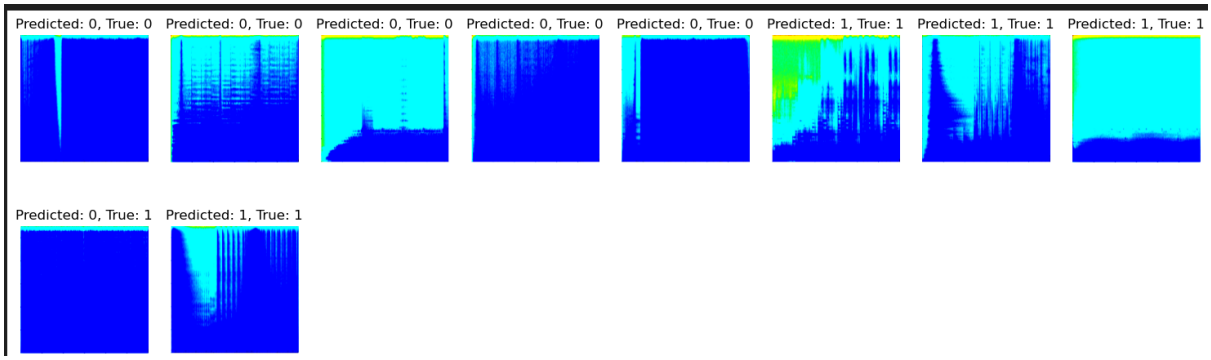


Figure 5 Visualized results of dataset "Is Epic Intro"

Method	Accuracy	Precision	Recall	F1-score	Runtime(s)
SVM	0.99	0.99	0.99	0.99	2m 39s
CNN	1.00	1.00	1.00	1.00	2m 12s
VGG16	1.00	1.00	1.00	1.00	10m 20s
MobileNet V3	1.00	1.00	1.00	1.00	1m 25s

Table 1 Result of Dataset "Is Epic Intro"

SVM: With an accuracy, precision, recall, and F1-score of 0.99, the SVM model performed exceptionally well. This suggests that the SIFT feature descriptors and the BoVW method provided a robust feature set for SVM to effectively find the decision boundary between the two classes.

CNN: The CNN achieved perfect scores across all metrics. Given that CNNs are particularly well-suited to capturing the hierarchical patterns in image data, this suggests that the model was able to learn discriminative features from the spectrogram's visual patterns to perfectly differentiate between epic and non-epic sections.

VGG16: Similar to CNN, the VGG16 model, with its deeper architecture, also achieved perfect scores. This could indicate that the depth of the VGG16, known for capturing intricate details in images, was able to generalize very well on this dataset, potentially capturing more nuanced differences in the spectrogram images.

MobileNet V3: This model scored slightly lower on all metrics compared to the other models. With a focus on efficiency and speed for mobile applications,

MobileNet V3 may not have captured as complex features as the other models. Nonetheless, a score of 0.90 and above is still indicative of good performance.

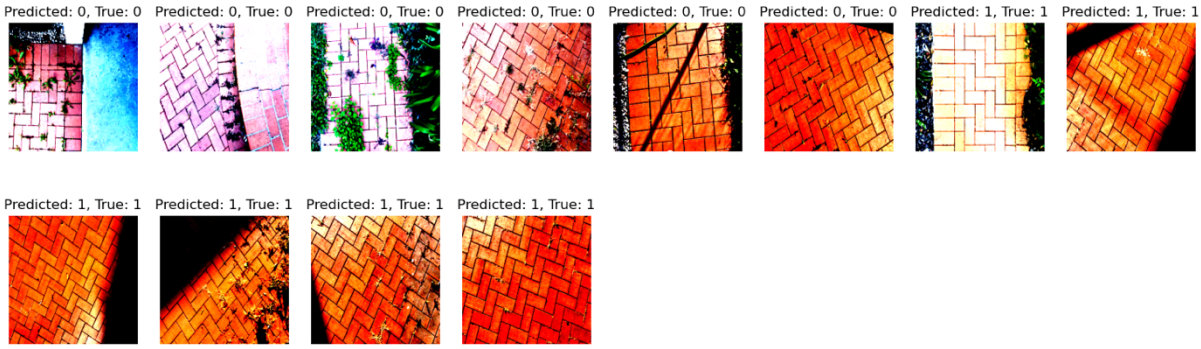


Figure 6 Visualized results of dataset "Needs Respray"

Method	Accuracy	Precision	Recall	F1-score	Runtime(s)
SVM	1.00	1.00	1.00	1.00	36m 56s
CNN	1.00	1.00	1.00	1.00	5m 31s
VGG16	0.92	0.93	0.92	0.92	12m 18s
MobileNet V3	1.00	1.00	1.00	1.00	5m 26s

Table 2 Result of Dataset “Needs Respray”

The results presented in Table 2 show the performance metrics for four machine learning models applied to the "Needs Respray" dataset:

SVM: Achieving perfect scores across all metrics, the SVM model has demonstrated an excellent capacity for distinguishing between live and dead weeds in the dataset. This suggests that the features extracted were highly discriminative and that the SVM's decision boundary effectively separated the two classes.

CNN: Similarly, the CNN model scored perfectly across all metrics. Given CNNs' capability for feature learning from image data, this indicates that the convolutional layers were successful in capturing the relevant patterns and textures indicative of live versus dead weeds.

VGG16: With slightly lower scores compared to the SVM and CNN models, the VGG16 still performed well with over 0.90 in all metrics. The reduction in performance relative to the other models could indicate that the VGG16

architecture, despite being deeper, may not have generalized as well to this specific task or that it may have required more fine-tuning of hyperparameters.

MobileNet V3: Matching the SVM and CNN in performance with perfect scores, MobileNet V3 demonstrates that even lightweight architectures, when well-tuned, can achieve high performance on specialized tasks such as this, balancing efficiency, and accuracy effectively.

Figure 7 Visualized results of model Custom CNN

Method	Accuracy	Precision	Recall	F1-score	Runtime(s)
SVM	-	-	-	-	-
CNN	0.95	0.95	0.95	0.95	7m 45s
VGG16	0.95	0.95	0.95	0.95	16m 23s
MobileNet V3	0.95	0.95	0.95	0.95	7m 16s

Table 3 Result of Dataset “Is Gen AI”

Table 3 displays the performance metrics for different machine learning models on the "Is Gen AI" dataset, which focuses on classifying images as either AI-generated or taken by a camera. CNN, VGG16, and MobileNet V3 – achieved an accuracy, precision, recall, and F1-score of 0.95, indicating highly effective classification with little variation in performance between the models. However, no results for SVM on the “Is Gen AI” dataset. Due to the traditional method of SVM caused memory overflow on our device. Given the dataset's objective of distinguishing AI-generated images from those taken by a camera, the

consistent performance across all models suggests that the features used for classification were robust and significant enough to differentiate between the two types of images. It's worth noting that AI-generated images may sometimes have tell-tale signs that distinguish them from photographs, such as patterns or textures that are atypical of natural scenes, or subtle anomalies in lighting or perspective.

Although models achieved nearly 100% accuracy on the validation set. However, analysis of loss values and performance on unseen test datasets revealed issues with overfitting. But, on the “Need respray” problem, it shows robust generalization ability, we think it’s because of its obvious features. Thus, In the future, we plan to incorporate more advanced feature extraction techniques to address this issue and further optimize model generalization.

5. Conclusion

Overall, Our research indicates that although traditional models like SVM and simple CNNs perform well under certain conditions, MobileNetV3 offers the best solution for practical applications requiring fast and efficient processing. MobileNetV3 achieves better accuracy compared to other models, especially on tasks like image classification and object detection. MobileNetV3 is also reduced latency which make it more efficient. Which makes it out most optimal choice of model.

However, when working with small datasets in machine learning, several challenges and issues may arise. Models are more prone to overfitting on small datasets, where they memorize the training data’s noise rather than capturing true patterns. To address these issues, there are some strategies. By applying transformations and enhancements to the training data, data augmentation increases diversity, mitigating overfitting issues. Transfer learning is another method. Utilizing models pre-trained on large datasets as initial model parameters, then fine-tuning them on the small dataset can accelerate model convergence and improve performance. Combining these strategies can help

overcome machine learning challenges on small datasets, improving model performance and generalization ability.

Ultimately, this research highlights the potential and practical significance of machine learning in various real-world applications, providing a roadmap for future innovations in the field. By carefully selecting models that align with specific task requirements and continuously adapting to technological advancements and data constraints, we can enhance the utility and impact of machine learning across diverse domains.

6. Reference

[1] Kulkarni, A., Shivananda, A., & Sharma, N. R. (2022). *Computer Vision Projects with Pytorch: Design and Develop Production-Grade Models* (1st ed.). Apress L. P. https://doi.org/10.1007/978-1-4842-8273-1

[2] Afifi, A. (2014). Laguerre Kernels –Based SVM for Image Classification. *International Journal of Advanced Computer Science & Applications*, *5*(1). https://doi.org/10.14569/IJACSA.2014.050103

[3] Koonce, B. (2021). *Convolutional neural networks with Swift for Tensorflow : image recognition and dataset categorization* (1st ed. 2021.). Apress. https://doi.org/10.1007/978-1-4842-6168-2

[4] Alshammari, A. (2022). Construction of VGG16 Convolution Neural Network (VGG16_CNN) Classifier with NestNet-Based Segmentation Paradigm for Brain Metastasis Classification. *Sensors (Basel, Switzerland)*, *22*(20), 8076-. https://doi.org/10.3390/s22208076