



# Tech4Nature México

## GENERAL AI + ENVIROMENT DOCUMENTATION



Tech4Nature



# Tech4NatureMéxico

## GENERAL AI + ENVIROMENT

### DOCUMENTATION

#### WHAT IS THE ROLE OF AI IN ENVIRONMENTAL WORK?

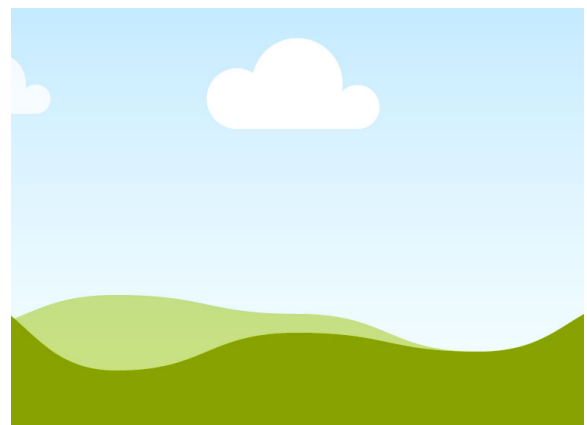
In our rapidly accelerating world, it is crucial that we address concerns about the well-being of the planet with the urgency they deserve. Environmental applications of emerging technologies present opportunities to do this. Among these technologies, artificial intelligence (AI) stands out as a powerful tool capable of revolutionizing our understanding, monitoring, and conservation efforts. However, we understand that AI can be complex and unfamiliar to many. With that in mind, this guide aims to provide non-technical users with a contextual overview of key AI terms within environmental applications. Our goal is to enable a broader audience to engage in meaningful discussions and contribute to the development of AI in this field.

#### WHAT ARE WE DOING?

**Tech4Nature Mexico** is a collaborative project that utilizes AI to support conservation efforts in the Yucatan Peninsula by monitoring priority ecosystems and species in the region. AI, specifically machine learning algorithms, is used to process images from camera monitoring technology to automatically detect and classify jaguars and, eminently, other species. Similarly, AI detects and classifies various bird and mammal species by using acoustic recordings. Given the extensive quantity of data from constant monitoring, these automated processes allow for more efficient monitoring than is manually possible, saving time and resources.

#### WHY ARE WE DOING IT?

AI is a complementary tool, not a replacement, for the work being done by real people in the environmental sector. While it may seem daunting, AI is an integral part of our current world and holds great potential to address environmental challenges more effectively. It is our responsibility to actively engage in the AI revolution and ensure its responsible use.



## GLOSSARY OF TECHNICAL TERMS

- **Algorithm:** A step-by-step procedure or set of rules followed to perform a specific task. A common algorithmic idea that we use in everyday life is sorting. When we sort items, (i.e. alphabetically), we can use a predetermined set of rules (i.e. the order of the alphabet) to systematically rearrange each item into its desired position.
- **Artificial Intelligence (AI):** The general branch of computer science that deals with the creation machines that perform tasks using algorithms that model human intelligence.
- **Machine Learning (ML):** A subset of AI that focuses on programs that learn from data to make decisions without explicit programming. Spam email filtering systems are a common example of ML. Programs compare the similarity of incoming emails to existing emails commonly marked as spam in order to determine legitimacy. As more incoming emails are marked as spam, programs learn more diverse representations of spam emails, which allow them to categorize future emails more accurately.
- **Deep Learning (DL):** A subset of machine learning that uses neural networks to learn high-level representations from complex data.
- **Neural Network:** a type of ML model that mimics the functioning of the human brain. Neural networks are composed of interconnected nodes, called neurons, which process and transmit information. The layered structure of neural networks enables them to capture complex relationships amongst the features of the data. For instance, in a water quality prediction neural network, different layers can represent features like temperature, pH, and nutrient levels. The extensive connections of neurons between layers allow the model to capture more nuances in the interplay of these features than a simple linear equation.
- **Convolutional Neural Networks (CNNs):** Neural networks specifically designed for processing grid-like data, such as images, to systematically detect patterns and features like patterns. For instance, in remote sensing, a CNN can be trained with various land cover images, enabling it to accurately classify different areas such as forests, crops, and urban regions by recognizing unique spatial arrangements and visual elements.
- **Training Data:** The data a machine learning model uses to learn patterns. Often, the data has corresponding labels, such as an image of a plant being labeled with its species name.
- **Preprocessing:** The initial data preparation step where data is cleaned, transformed, and normalized to make it suitable for analysis or training.

## Tech4NatureMéxico

- **Data Augmentation:** Techniques used to increase the size and diversity of a training dataset by applying various transformations. It improves the ability of AI models to generalize their performance to unseen data. For instance, in climate modeling, data augmentation can involve generating synthetic climate scenarios to simulate future climate conditions for impact assessments.
- **Supervised Learning:** A type of machine learning where the model learns from labeled training data, associating input data with correct outputs to make predictions or classifications. For example, in supervised learning, a model could utilize a training dataset of labeled images of jaguars and non-jaguars to learn common visual features that distinguish a jaguar from other animals.
- **Unsupervised Learning:** A type of machine learning where the model learns "inherent" patterns from unlabeled training data, aiming to discover hidden structures or clusters. Unsupervised learning could be used to identify clusters of similar species based on their ecological traits, helping scientists discover new relationships and understand ecosystem dynamics beyond predefined classifications.
- **Feature Extraction:** The process of automatically extracting relevant features from raw data. In a program that analyzes whale vocalizations, each recording contains a mix of different environmental noises and interference. Features extraction could isolate the duration of specific calls or the frequency range of vocalizations.
- **Computer Vision:** The field of AI that enables computers to interpret and understand visual information from images.
- **Object Detection:** Within computer vision, the task of locating and classifying objects in an image. For example, in forestry, object detection can be used to identify tree species from aerial images, assisting in forest inventory and management.
- **Audio Classification:** The task of categorizing audio signals into predefined classes or categories. In ornithology, audio classification can be applied to identify bird species based on their unique vocalizations, helping monitor bird populations and biodiversity.
- **Transfer Learning:** A technique in machine learning where knowledge or representations learned from one task are applied to another related task, saving time and computational resources while achieving accurate results for new objectives. For instance, a model trained to identify plant species in one region could be fine-tuned and applied to identify plant species in a different region with similar flora.

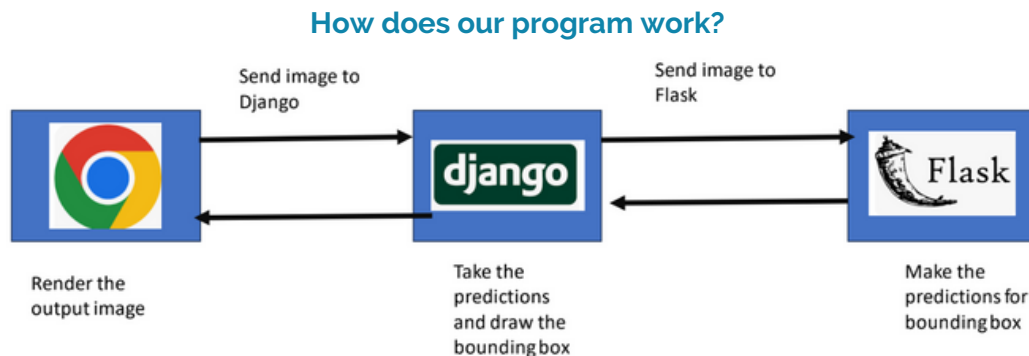
- **Ethics in AI:** the principles and guidelines that govern the responsible and ethical development, deployment, and use of artificial intelligence systems—avoiding harm to individuals, society, and the environment ([resources](#))

## SPECIFIC DOCUMENTATION

### Object Detection

#### What is object detection?

Object detection programs analyze an image to predict the presence of an “object of interest” and its location within the image. In this context, our objects of interest are jaguars and other animals. Since constant camera-trap monitoring produces thousands of images, many of these will contain no wildlife. Sometimes, even if wildlife is present, it can be hard to manually identify due to camouflage, image quality, and many other reasons. It can greatly save time and improve accuracy to run batches of images through an object detection program that is trained to account for these factors.

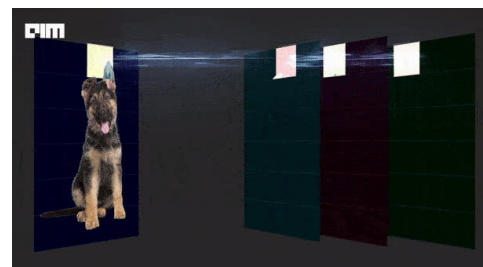


#### 1. Upload and Process Photo(s)

To begin, upload photo(s) to the web application from your device. The web application is built using Django, which can be thought of as a library of valuable tools that are often used in web applications. Once the web application receives the photo(s) data, it will preprocess the uploaded photo(s) (e.g. resize the image) to ensure it is in an appropriate format for our object detection models.

#### 2. Run Object Detection Models on the Photo(s)

Now the processed photos are sent to our object detection models, which are hosted using Flask, another library with helpful tools for web applications. The models used by our program are called Convolutional Neural Networks (CNNs), which are a type of model trained to process visual data.



CNNs scan the photo using small filters, like magnifying glasses, to look for distinct visual patterns, such as shapes, textures, and edges. These filters create feature maps that highlight the presence of these patterns in different regions of the image. CNNs then apply different operations to process the data from feature maps in a meaningful way. For example, if a filter is trained to produce a feature map that highlights the presence of leaf-like textures within an image, an operation might then be applied to de-emphasize the regions where leaf-like textures are present since it is trying to locate wildlife, not plants.

After applying various filters, the CNN will have enough information to predict the most important region of the photo. If it decides that a region is important enough to likely contain a wildlife object, it will return the region's "bounding box", the image coordinates of the rectangular frame around the detected animal.

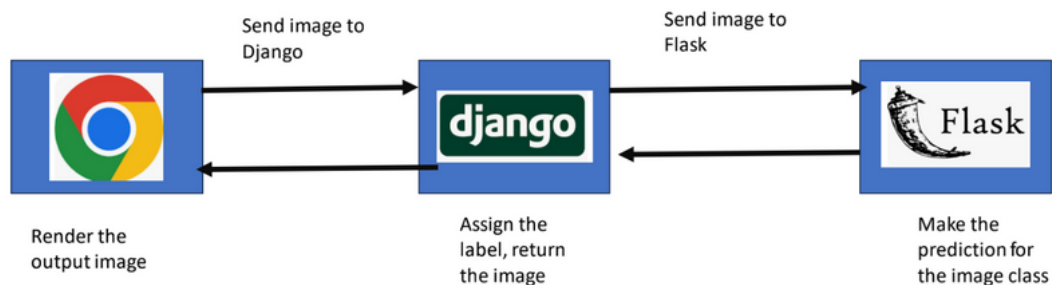
### 3. Return the Resulting Image

The model then returns the predicted image coordinates (or lack thereof) of the bounding box back to the web application. The web application then renders a bounding box on top of the image and returns the updated image to the user.

### What is image classification?

Image classification programs assign a category to an image. They do this by comparing the image to existing images that have already been categorized. In this context, the program can predict the presence of a jaguar by determining whether the image is more similar to those in the "jaguar present" or "jaguar not present" category. When monitoring a specific species, an image classification program can greatly increase efficiency by automatically identifying the most relevant images.

### How does our program work?



## 1. Train the Image Classification Model

The models used by our program are called Convolutional Neural Networks (CNNs), which are a type of model trained to process visual data. First, the CNN is trained on a large dataset of labeled images, with each image labeled with the corresponding species category (e.g. jaguar, no-jaguar, etc.). The CNN scans each photo using small filters, like magnifying glasses, to look for distinct visual patterns, such as shapes, textures, and edges. The CNN learns from these labeled examples to recognize the unique features and patterns associated with each species. Once the image classification model is trained, it can be used for inference during wildlife detection.

## 2. Upload and Process Photo(s)

To begin, upload photo(s) to the web application from your device. The web application is built using Django, which can be thought of as a library of valuable tools that are often used in web applications. Once the web application receives the photo(s) data, it will preprocess the uploaded photo(s) (e.g. resize the image) to ensure it is in an appropriate format for our image classification models.

## 3. Run Image Classification Models on the Photo(s)

Now the processed photos are sent to our object detection models, which are hosted using Flask, another library with helpful tools for web applications. Similarly to the training phase, during the detection phase, the CNN scans the photo with small filters to identify visual patterns. It compares the detected patterns to the learned patterns of the labeled training images to determine which class the image belongs to. It returns this prediction to the web application.

## 4. Return the Resulting Image

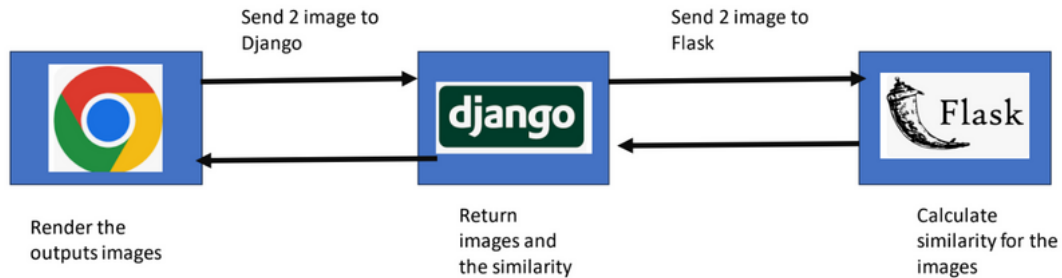
The web application then displays the labeled image to the user.

## Individuality Detection

### What is individuality detection?

Individuality detection programs compare the objects of interest of multiple images to determine whether or not they are unique objects or multiple instances of the same object. They do this by comparing the similarity of visual features between the two objects. In this context, the program can take multiple images of jaguars and compare features such as the patterns of their coats to determine if they are different jaguars. Since camera-trap monitoring produces images of animals from many angles over long periods of time, determining the individuality of the animals within these images helps understand population dynamics and behaviors more accurately.

## How does our program work?



### 1. Train the Siamese Network

Our program is a specialized model called a Siamese Network, which is a model often used in facial recognition systems. This network is trained to identify individual jaguars based on their unique features. During the training phase, the Siamese Network learns from a dataset of labeled images, where each image is paired with another image and labeled as either "same individual" or "different individuals." The network employs identical sub-networks which each learn the features of one input image—the identicalness of the sub-networks ensures that two extremely similar images (e.g. images of the same individual) could not be determined to have very features because each network computes the same function. The network then learns to group images, where similar images are closer together and dissimilar images are farther apart.

### 2. Upload and Process Jaguars Images

To begin, upload the photos of the jaguars that you would like to compare the similarity of to the web application from your device. The web application is built using Django, which can be thought of as a library of valuable tools that are often used in web applications. Once the web application receives the photo(s) data, our program begins to preprocess them.

Distinguishing the individuality of jaguars often is determined by very small details—the general similarity of jaguar colors and environment makes this task challenging. It helps to focus only on the pixels belonging to the jaguar of interest. To address this, the program utilizes an advanced AI model called Segment Anything Model (SAM), which first specifies the object's location using bounding boxes, eliminates the background by setting its pixels to white, and then crops the image using the bounding box coordinates. Now, any irrelevant data from the background, such as shadows, trees, sticks, and leaves, have been removed. This preprocessed image is then ready to be passed through the Siamese Network for similarity calculation.



### 3. Image Similarity Calculation with Siamese Network

The preprocessed jaguar images are then passed to our object detection model, which is hosted using Flask. The Siamese Network takes the two images as input. Each image is processed separately through its twin sub-network to extract their feature representations in the high-dimensional space.

The Siamese Network calculates the similarity between the two jaguar images based on their feature representations. A high similarity score indicates that the images are likely of the same individual jaguar, while a low score suggests they belong to different individuals.

### 4. Returning the Similarity Result

Finally, the web application displays the similarity score to you, along with an explanation of the result. You'll see whether the uploaded images are likely of the same individual jaguar or represent different jaguar individuals.