Project Design Phase-I
Solution Architecture

Date	19 September 2022
Team ID	PNT2022TMID29650
Project Name	Project – Digital Naturalist AI enabled tool for diversity researchers
Maximum Marks	4 Marks

Solution Architecture:

Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

- Find the best tech solution to solve existing business problems.
- Describe the structure, characteristics, behaviour, and other aspects of the software to project stakeholders.
- Define features, development phases, and solution requirements.
- Provide specifications according to which the solution is defined, managed, and delivered.

ABOUT:

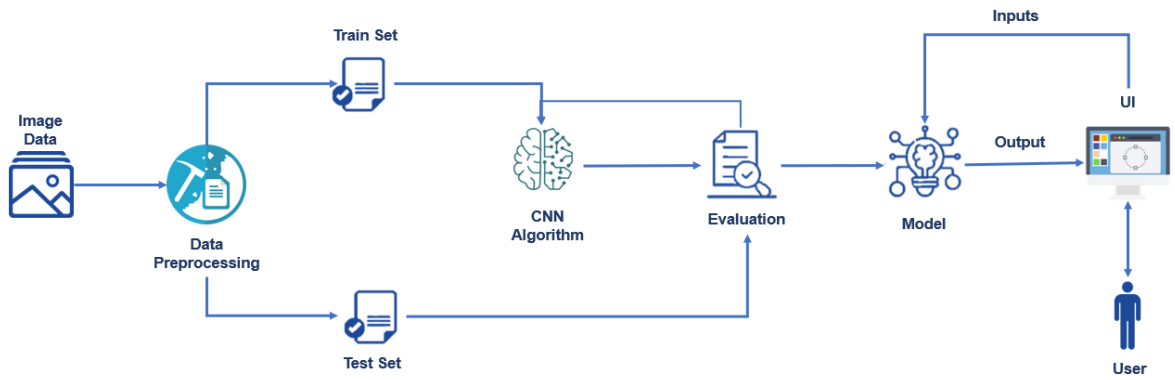
A naturalist is someone who studies the patterns of nature, identifies a different kind of flora and fauna in nature.

Which leads to an interest in protecting wild spaces, and collecting and sharing information about the species we see on our travels is very useful for conservation groups like NCC.

Project:

A project where we are creating a web application which uses a deep learning model, trained on different species and get the prediction of the bird when an image is been given.

TECHNICAL ARCHITECTURE:



- Download/create dataset, Augment the dataset to virtually increase the size of small datasets in order to make our machine learning models work better.
- Pre-process the images to a machine-readable format and load the data into Numpy Arrays
- Perform a Train Test Split on the dataset.
- Define the model creation function: adding all the neural network layers required.
- Fit the model on train data and check for accuracies using test data as well.
- Save the model and its dependencies.
- Build a Web application using flask that integrates with the model built.
- Apply CNN (Convolutional Neural Network) algorithm on the dataset, where the deep neural networks predict the class and subclass of a given image.
- Build web applications using the Flask framework.

TOOLS REQUIRED:

1. **ANACONDA NAVIGATOR** – For data science and machine learning related applications
2. **DEEP LEARNING MODELS:**
    - Tensor flow (To create and deploy Machine learning tools and applications)
    - Keras (To make high level neural network API, user friendly, high scalable computation)
    - Flask (To build Web Applications)
3. **GitHub** to load the dataset and data augmentation.
4. IBM Cloud and IBM Watson.

PROCESS:

1. Data Collection:

Collecting datasets from different open sources like kaggle.com, data.gov, UCI machine learning repository, etc.

Link to download the DATASET:

https://drive.google.com/file/d/1hjPKoJi-3t0yZJnPoJF7gNAMn0rgIXcr/view

2. DATA AUGUMENTATION, LOADING DATA AND PREPROCESSING:

Augmenting using **os. mkdir()** function in python or manually and change the directory paths to load and save.

Use keras for image pre-processing package which is Image Data Generator. Then save or load all the images in path and call function for each sub folder by giving input params.

Link to Load and pre-process the data:

https://thesmartbridge.com/documents/spsaimldocs/CNNprep.pdf

3. DATA SPLITTING INTO Train AND Test:

a) Convert data into Numpy arrays to feed it to the model.
b) Convert the Y data to categorial using keras inbuilt function which works almost like One Hot Encoder.
c) Split the data into Test and Train.

4. CONVOLUTION NEURAL NETWORKS:

NEURAL NETWORKS:

A Tool to teach the computer to think and learn by classifying information, similar to humans' way of -learning, predicting, thinking, recognizing.

DEEP LEARNING:

Focuses on machine learning tools to solve and make decisions. And, processes the data through neural networks

- NumPy
- Object Detection
- Template Matching
- Corner, Edge, and Grid Detection
- Feature Matching
- Object Tracking
- Deep Learning with Keras
- Keras and Convolutional Networks
- Customized Deep Learning Networks

CONVOLUTION NEURAL NETWORKS:

A type of advanced artificial neural network. It consists of

- An Input layers.
- An Output layers.
- Multiple hidden layers.

The **hidden layers of a CNN** typically consist of

- convolutional layers
- pooling layers
- fully connected layers
- normalization layers. (**Flatten** layer is used between the convolutional layers and the dense layer to reduce the feature maps to a single one-dimensional vector.)

Additional Layers:

- **Strides (**weight matrix that indicates Number of Jump in the input layer**)**
- **Padding (**Amount of pixels added to an image while processing by the kernel of CNN**)**

- **Max Pooling (**Selects maximum elements from the region of feature map covered by filter**)**
- **Dropout Layer (** Removes inputs – input variables in the data sample or activations from previous layer, from a layer**)**
- **fully connected layers** (recognize and classify images for computer vision. layers where all the inputs from one layer are connected to every activation unit of the next layer.)
- **Flatten layer** (used between the convolutional layers and the dense layer to reduce the feature maps to a single one-dimensional vector.)

5. Building Model

- Summary - model.summary() is used to see all parameters and shapes in each layer in our model.
- Compile
- Fit - After compiling our model, we will train our model by the fit() method, then evaluate it.
- Predict

6. EVALUATION, MODEL SAVING AND LOADNG, BUILD APPLICATION:

Step 1: EVALUATION

- Accuracy - Used to measure the algorithm's performance in an interpretable way. Is usually determined after the model parameters and is calculated in the form of a percentage.
- Losses - How poorly or well a model behaves after each iteration of optimization.
- Load a Test Image and Make a Test Prediction

Step 2: To save model and load it: Saving The Model (get_weights, set_weights), this can take weeks to train.

Step 3: Application building using Flask

- Load the required packages
- Initialize graph, load the model, initialize the flask app and load the model Graph elements to work with tensorflow.
- Configure the home page
- Pre-process the frame and run

Link to Develop the application:

https://www.youtube.com/watch?v=lj4I_CvBnt0

7. BUILD THE HTML Page AND EXECUTE:

- Run the application
- Open the browser and navigate to localhost:5000 to check your application

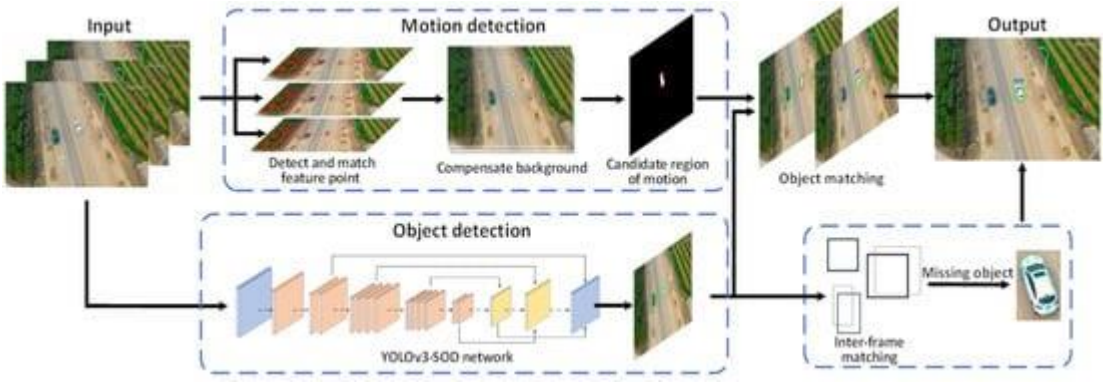
8. TRAIN THE MODEL ON IBM:

By registering for IBM Cloud and then Train the model on IBM Watson.

Other Tech solutions that can be used to enhance the process:

1. To Secure the storage of Users Data
    - Using cloud storage to back up personal data like
        - ❖ One Drive
        - ❖ Drop Box
        - ❖ iCloud
        - ❖ Strath cloud
    - Providing password protection to protect the data stored in cloud
    - Which ensures Backup of user's data even if the device seem to be lost
    - The One Drive and Strath cloud stores the data securely On-Site with data replicated between multiple data centres without the need to make own backups as data is backed up automatically.

2. To get good recognition of moving objects



- **Motion detection:**

Obtain the rough motion position with the background compensation method.:

Detection using 4 algorithms are:

- ❖ FAST
- ❖ Harris
- ❖ SURF
- ❖ SIFT

Based on the coordinate relationships of the matched feature points, we compare the detection performance of several feature point detections algorithms and selected the **SIFT algorithm**.

The **SIFT algorithm** is a feature point detection algorithm based on scale space, which exhibits good invariability with respect to image translation, rotation, scaling, shielding, noise, and brightness changes.

This algorithm creates a descriptor for each feature point, which is represented by a 128-dimensional vector, and SIFT feature points with minimum Euclidean distance are matched in two adjacent frames.

Problems associated with the inter-frame difference method:

- Shadow
- Reflection
- Occlusion
- Inaccurate registration.

So, we apply CNN to solve these problems

- **Object detection:**

  YOLOv3-Small Objects Detection (SOD) can be used to detect the locations of objects.

  - ❖ 2 improvements to YOLOv3 for the moving object detection task.

  i. Considering the possibility of deploying an UAV and surveillance camera, it is necessary to further reduce the size of the model and improve the speed of the model.
  ii. The moving object detection task involves a large number of small moving targets. We counted the numbers of pixels for the moving targets in the images and found that the targets with 20–50 pixels comprised the majority; therefore, the YOLOV3 network needs to be optimized for small-target detection. Thus, we introduce the YOLOv3-SOD network

  - ❖ In the CNNs, the deep layers contain semantic information, whereas the shallow layers contain position information.

  - ❖ The small object features may be lost as the number of network layers deepens, so YOLOv3 uses up-sampling to merge the different layers of the feature maps to solve this problem.

  - ❖ The same method as YOLOv3 is used in YOLOv3-SOD to improve the ability to detect small targets.

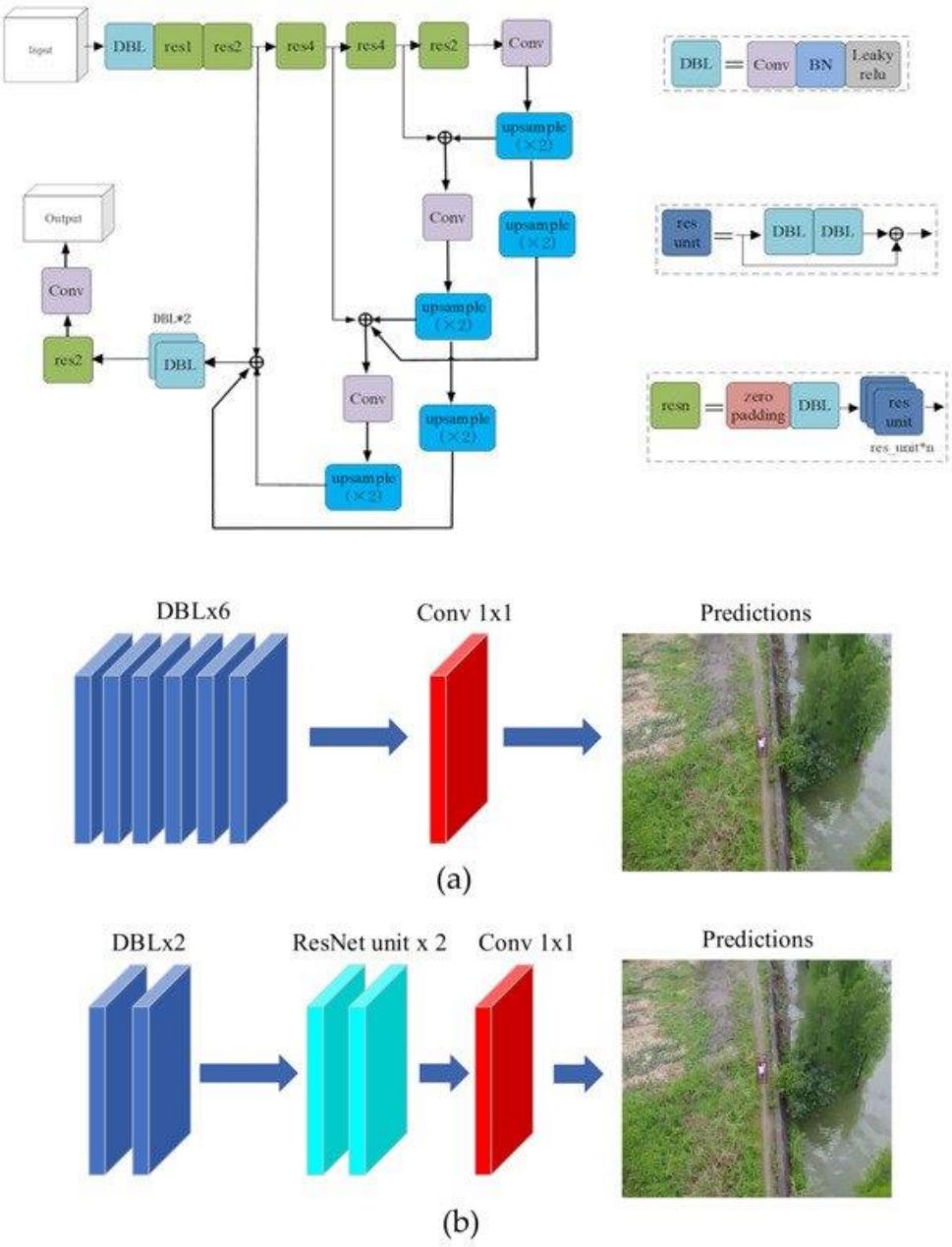


*Fig: Structure of the output layers for: (**a**) YOLOv3; and (**b**) YOLOv3-SOD.*

- **Object matching:**

A moving object is determined by matching the motion detection and object detection results, while temporal and spatial types of information from adjacent frames are used to recall the missing detections.

We use intersection over union (*IoU*) to unify the two sets of detection results.

The bounding box for **motion detection** is assumed to be M and the bounding box for **object detection** is *D*, so the formula for calculating *IoU* is as follows.

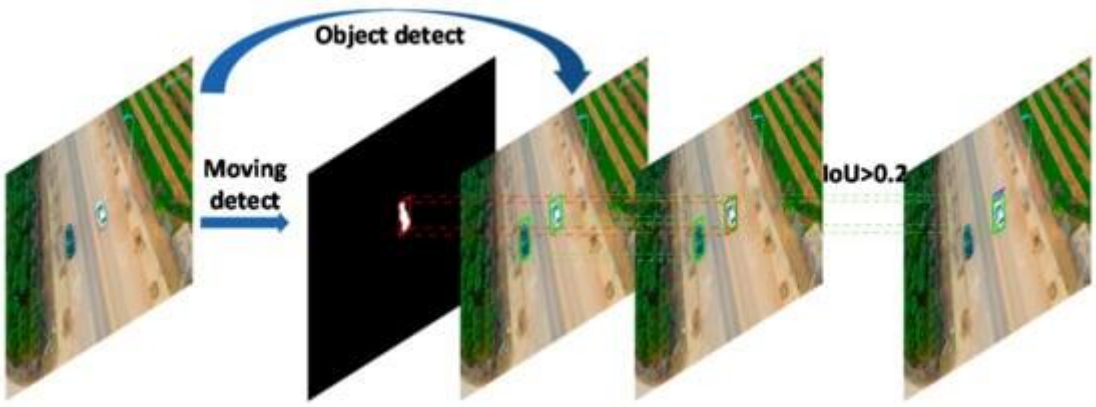
$$\mathrm{IoU} = \frac{\mathrm{area(M)} \cap \mathrm{area(D)}}{\mathrm{area(M)} \cup \mathrm{area(D)}}$$

For, m objects in the image with n moving objects, where Di is the object detection result for object i, i ϵ (0, m], and Mj is the motion detection result for moving object j, j ϵ (0, n].

For object i,

Moving Object = Di if max (IoU(Di,Mj)) > 0.2,jϵ(0,n]

Fig: Method for determining a moving object.



REFERENCES:

- https://careereducation.smartinternz.com/saas-guided-project/3/digital-naturalist-ai-enabled-tool-for-biodiversity-researchers
- https://www.strath.ac.uk/professionalservices/is/cybersecurity/storingdatasecurely/
- https://www.mdpi.com/2073-8994/12/12/1965/htm