

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

BỘ MÔN THỰC TẬP CƠ SỞ



## THỰC TẬP CƠ SỞ

<b>Giảng viên hướng dẫn</b>	<b>: KIM NGỌC BÁCH</b>
<b>Họ và tên sinh viên</b>	<b>: NGUYỄN XUÂN HẢI</b>
<b>Mã sinh viên</b>	<b>: B22DCCN271</b>
<b>Lớp</b>	<b>: D22CQCN07-B</b>
<b>Nhóm</b>	<b>: 13</b>

*Báo Cáo Hàng Tuần Lần 4 ngày (30/3-5/4/2025)*

*Hà Nội – 2025*

# I. NHIỆM VỤ CỦA TUẦN NÀY

## 1. Thu thập dữ liệu và tiền xử lý

**Animal Species Classification - V3** 63 <> Code Download

Data Card Code (25) Discussion (1) Suggestions (0)

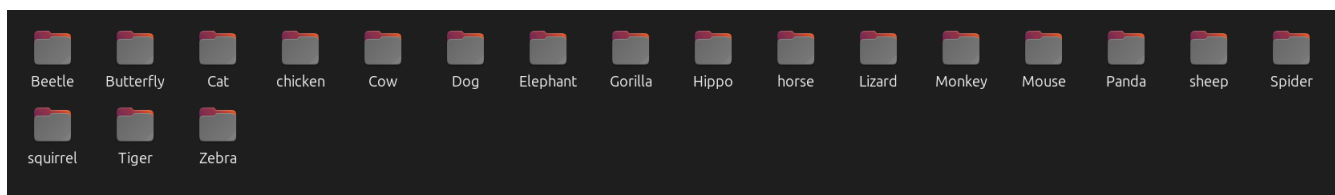
### Class Distribution

This is **Animal Classification Data-set** made for the **Multi-Class Image Recognition Task**. The dataset contains **15 Classes**, these classes are :

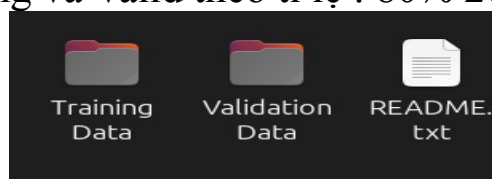
1. Beetle
2. Butterfly
3. Cat
4. Cow
5. Dog
6. Elephant
7. Gorilla
8. Hippo
9. Lizard
10. Monkey
11. Mouse
12. Panda
13. Spider
14. Tiger
15. Zebra

Tags: Computer Science, Biology, Image, Classification, Computer Vision, Animals, Multiclass Classification

- Tải bộ dữ liệu Animal Species Classification - V3 trên KAGGLE
- Gồm có 15 loại động vật: bọ hung(Beetle), bướm(Butterfly), mèo(Cat), bò(Cow), chó(Dog), voi(Elephant), khỉ đột(Gorilla), hà mã(Hippo), thằn lằn(Lizard), khỉ(Monkey), chuột(Mouse), gấu(Panda), nhện(Spider), hổ(Tiger), ngựa vằn(Zebra)
- Tải bộ dữ liệu trên các mã nguồn mở để lấy các bộ dataset loài động vật quen thuộc như gà, ngựa, cừu, sóc



- Chia thành 2 bộ Training và Valid theo tỉ lệ : 80% 20%



## 2. Tạo Repository trên github

### Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

---

*Required fields are marked with an asterisk (\*).*

Owner \*


Nxhaicr7

 /


Great repository names are short and memorable. Need inspiration? How about **fluffy-octo-telegram** ?

Description (optional)

---

☒  Public

Anyone on the internet can see this repository. You choose who can commit.

☐  Private

You choose who can see and commit to this repository.

---

Initialize this repository with:

☐ Add a README file

This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore

.gitignore template: None

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

## 3. Tiền xử lý dữ liệu

### - Tạo bộ Animals

```
class Animals(Dataset): 4 usages
    def __init__(self, root_path, is_train = True, transform = None):
        self.transform = transform
        if is_train:
            data_paths = os.path.join(root_path, "Training")
        else:
            data_paths = os.path.join(root_path, "Validation")
        self.labels = []
        self.images = []
        self.catelogies = ['Beetle', 'Butterfly', 'Cat', 'Chicken', 'Cow',
                           'Dog', 'Elephant', 'Gorilla', 'Hippo', 'Horse',
                           'Lizard', 'Monkey', 'Mouse', 'Panda', 'Sheep',
                           'Spider', 'Squirrel', 'Tiger', 'Zebra']

        for ind, catelogy in enumerate(self.catelogies):
            catelogy_path = os.path.join(data_paths, catelogy)
            for file in os.listdir(catelogy_path):
                file_path = os.path.join(catelogy_path, file)
                self.images.append(file_path)
                self.labels.append(ind)

    def __len__(self):
        return len(self.images)

    def __getitem__(self, item):
        image_path = self.images[item]
        image = cv2.imread(image_path)
        print(image_path)
        if self.transform:
            image = self.transform(image)
        label = self.labels[item]
        return image, label
```

**\_\_init\_\_(): Load ảnh và nhãn từ thư mục.**

- Nhận 3 siêu tham số: đường dẫn đến file(root\_path), xác định lấy file train hay valid(is\_train) , lưu pipeline biến đổi ảnh(transform)
- self.labels: Danh sách chứa nhãn của ảnh.
- self.images: Danh sách chứa đường dẫn ảnh.
- self.catelogies: Danh sách tên các lớp, với mỗi lớp có một chỉ mục từ 0 đến 18.

**\_\_len\_\_(): Trả về số lượng ảnh.**

**\_\_getitem\_\_(): Lấy một ảnh và nhãn theo index.**

- Chỉ số ảnh cần lấy: Item.
- Biến đổi ảnh(transform) rồi trả về ảnh và nhãn.

## - Tiền xử lí dữ liệu

```
if __name__ == "__main__":
    transform = Compose([
        ToTensor(),          #Transform to Tensor
        Resize((526, 526)) #Resize image
    ])
    data = Animals( root_path: "/home/nxhai/Downloads/AI/ComputerVision/ComputerVision/animals", is_train: True, transform) #Choose training image
    test_image, test_label = data[0]
    label = data.catelogies[test_label]
    test_image = test_image.permute(1, 2, 0).numpy() #Convert tensor(C,H,W) to NumPy(H,W,C) for display with OpenCV
    test_image = (test_image * 255).astype(np.uint8) #Return [0,1] to [0,255]
    cv2.imshow("{}".format(label), test_image)
    cv2.waitKey(0)
```

- Chuyển ảnh về dạng Tensor(): Chuẩn hóa về dạng pixel [0,1] dạng (C,H,W) (giúp dễ dàng chuyển đổi ảnh thành định dạng phù hợp cho PyTorch, giúp mô hình hoạt động tốt hơn)
- Chuyển tất cả kích cỡ ảnh về dạng chuẩn

**\* Thêm: Nếu muốn trình diễn ảnh thì chuyển về dạng Numpy để dùng thư viện openCV hiển thị**