CATTLE IDENTIFICATION USING MUZZLE PATTERN RECOGNITION

Abstract:
The ability to identify specific cows is a critical aspect of keeping cattle. Registration is essential for the production, distribution, and breeding of cows. It is dangerous to brand cows in the ear in the traditional manner. To solve this issue, we recommend using a biometric scanner for identification. Numerous studies have shown that the muzzle pattern can be used to distinguish animals and eliminate inconsistent patterns. On the tip of their nose or muzzle, the majority of animals have a distinctive pattern. Animals can be recognized by muzzle patterns, just like with fingerprints of human. Systems for animal identification are essential, as they play a crucial role in various applications such as processing loan and insurance requests, as well as facilitating medical interventions and treatments. Using a biometric muzzle scanner, the approach of machine learning, image processing, and computer vision has been used for identification purposes.

Existing Method:
Traditional identification methods like **ear tags and tattoos**, which could harm cows and become damaged over time.

Dataset Description:
- Collected images of cow's nose from six cows nearby my home.
- Performed Data Augmentation and made a dataset of 132 images in total.
    - Training - 114 images of six cows
    - Validation – 18 images of six cows

Proposed Method:

Initially gathering real-time cow muzzle images and processing them through data preprocessing, augmentation, and segmentation to isolate the nose area. Utilizing the **Scale-Invariant Feature Transform (SIFT) algorithm**, distinctive features are extracted from these segmented nose images. These features are encoded with unique IDs using the MD5 Hashing algorithm to prevent redundancy. The subsequent step involves comparing these features between test and training images of the same cow for pattern matching, enabling accurate identification of unique nose patterns. Validation, of results are then carried out to ensure accuracy.

Screenshots:

- **Feature Extraction:**

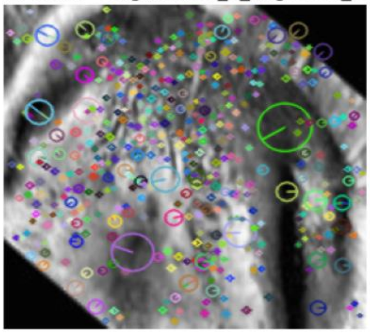
```
# Add New Cow
name_of_cow = input("Enter the cow name: ")

# Define the path to the folder for cow
cow_folder_path = '/content/drive/MyDrive/Colab Notebooks/Muzzle recognition/CowDataset/Data/training/cow1'

extract_features(name_of_cow, cow_folder_path)
```

```
Enter the cow name: cow1
```

Keypoints for cow1 - augmented_0_angle150_scale1.2.jpg



Keypoints for cow1 - augmented_0_angle120_scale1.5.jpg

- **Features**

```
print(cow_features)
```

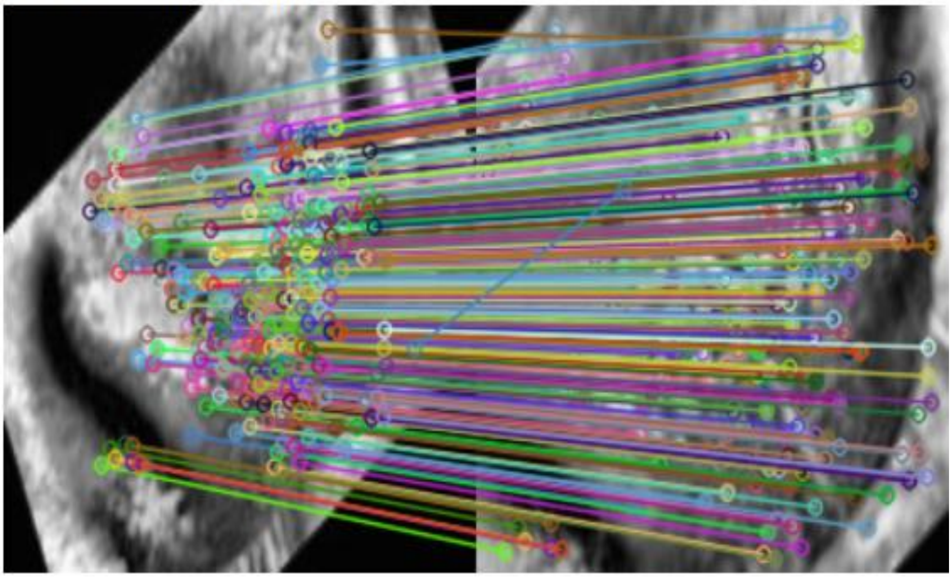
```
{'cow1': {'features': [array([[ 22.,   3.,   0., ...,   1.,   0.,   2.],
        [ 28.,  78.,   7., ...,   0.,   0.,   1.],
        [194.,  21.,   0., ...,   0.,   0.,   0.],
        ...,
        [  6.,   5.,   0., ...,   8.,   0.,   0.],
        [ 26.,  28.,  11., ...,   1.,   8.,   3.],
        [  0.,   0.,   0., ...,  12.,   8.,   3.]], dtype=float32), array([[  0.,   0.,   0., ...,   0.,  12., 143.],
        [  0.,   0.,   0., ...,   2.,   0.,   0.],
        [  0.,   0.,   0., ...,   2.,  17.,  39.],
        ...,
        [  0.,   0., 116., ...,  70.,   0.,   0.],
        [  9.,   4.,  12., ...,   0.,   0.,   4.],
        [  0.,   0.,   0., ...,   3.,   4.,   3.]], dtype=float32), array([[ 68.,   1.,   0., ...,  19.,   3.,   0.],
        [  5.,   2.,   1., ...,   0.,   0.,   0.],
        [174.,   3.,   0., ...,  29.,  19.,   0.],
        ...,
        [ 48.,   0.,   0., ...,   0.,   3.,   5.],
        [  1.,   0.,   0., ...,  26.,   2.,   6.],
        [  1.,   0.,   0., ...,  13.,   1.,   2.]], dtype=float32), array([[63.,  2.,  0., ...,  9., 10., 43.],
        [35.,  1.,  0., ..., 20.,  0.,  0.],
        [56.,  3.,  0., ...,  3.,  3., 13.],
        ...,
        [40.,  2.,  0., ...,  0.,  1.,  2.],
        [ 1.,  0.,  0., ..., 34.,  2.,  3.],
        [ 1.,  0.,  0., ..., 17.,  1.,  1.]], dtype=float32), array([[141., 141.,   1., ...,   7.,   0.,   1.],
        [ 43., 136.,  41., ...,   0.,   3.,   4.],
        [109.,  19.,   5., ...,   1.,  25.,   5.],
        ...,
        [129., 129.,  16., ...,   0.,   0.,   4.],
```

Result for Cow1:

- **Test Image 1:**

254

0.42474916387959866



- **Test Image 2:**

262

0.5887640449438202