implicitShapeModelDetectionTask

May 6, 2025

1 Detection: Test Your Model on Images Showing Objects in Background

1.0.1 Pipeline

1. Load & display images
2. Extract ORB keypoints/descriptors
3. Visualise **all** raw keypoints (red)
4. Apply NMS
5. Visualise **retained** keypoints (green)
6. Save the NMS-pruned descriptors (`.npy`) and keypoint metadata (`.csv`)

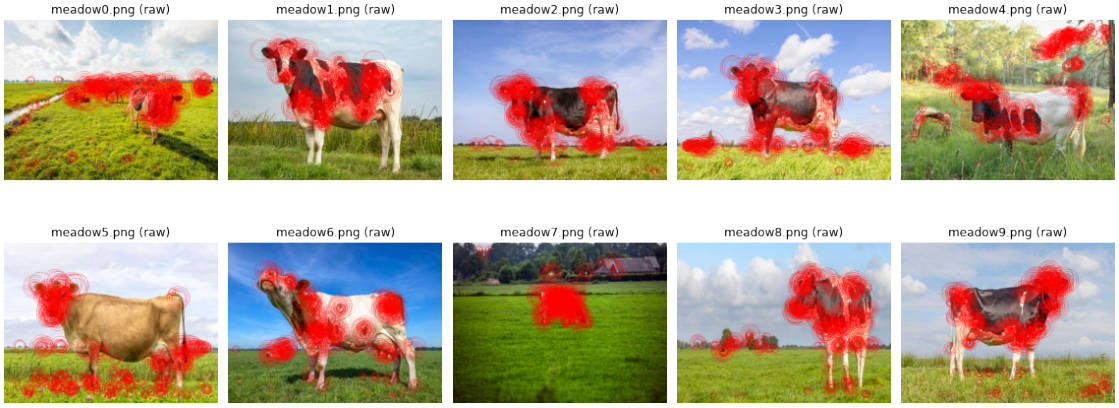
Dependencies: `opencv-python`, `numpy`, `matplotlib`.

[1]:

[2]:

```
Keypoints per image (raw): [1000, 1000, 1000, 1000, 1000, 1000, 1000, 1000,
1000, 1000]
```

[3]:



[4]:

```
Keypoints per image (after NMS): [99, 40, 53, 60, 62, 97, 67, 26, 55, 91]
```

[5]:

[6]:

```
Saved pruned features to /media/hellwich/Herakles/hellwich/hellwich/lehre/Aut_BA
/2025Exercises/Exercise2/orb_features_nms
```

1.1 Load object & background cluster models

Load model data: - **Object** clusters: `orb_allfeatures_cow/cluster_centers.npy` and their displacement lists `cluster_displacements.json`. - **Background** clusters: `orb_allfeatures_background_images/cluster_centers.npy`.

Executed once before processing test images.

[7]:
```
# 16. Load object & background cluster models¶
```

```
Loaded 16 cow and 27 background clusters
```

1.2 Assign test keypoint feature descriptors to nearest cluster (codebook entry / visual word) and keep object keypoints only

[8]:
```
# 17. Assign test keypoint feature descriptorss to nearest cluster (codebook␣
↪entry / visual word) and keep object keypoints only
```

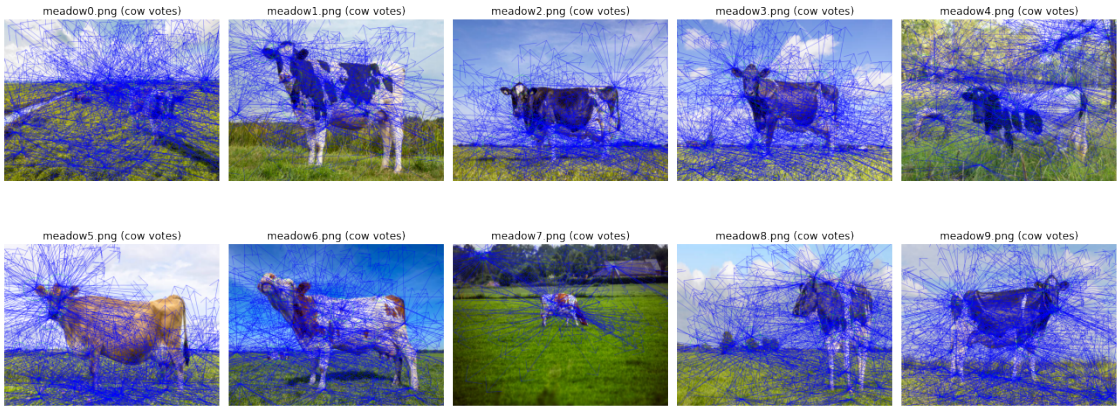
```
Cow keypoints kept per image: [41, 22, 28, 34, 33, 36, 30, 9, 35, 44]
```

1.3 Voting procedure

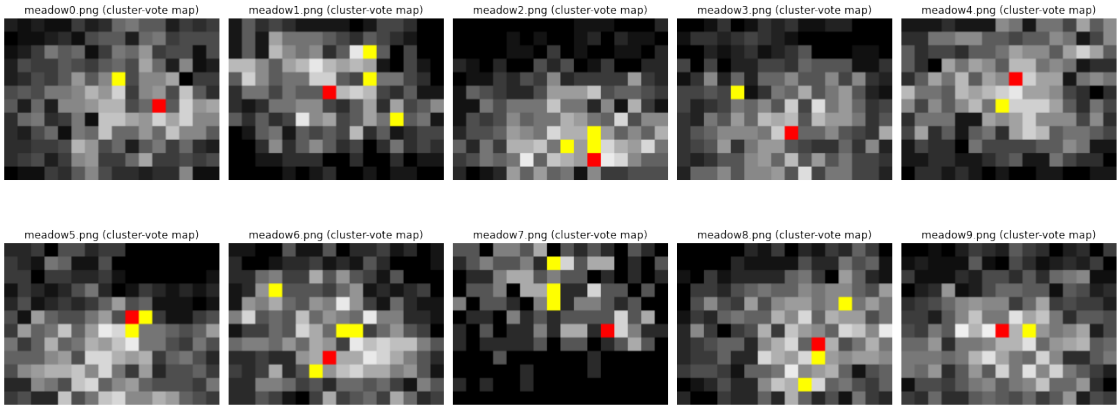
Visualise object votes (displacement vectors from cluster).

Here it is done on the test images and not on the training images only. So this is the application of the Implicit Shape Model to real data.

[9]:
```
# 18. Visualise object votes (same as in training jupyter notebook)
```



[10]:
```
# 19. Cast votes to low resolution grid bins and visualize (same as in training␣
↪jupyter notebook)
```



2 Discussion

Discuss the achievements of you **prototype** Implicit Shape Model:

- Are the result reasonable in principle?
- What are the deficits?
- Suggest improvements:
  - Does your approach make use of an angle or a direction of the keypoint?
  - If not, what would you need to change?
  - Does your approach make use of a size or scale of the keypoint?
  - If not, what would you need to change?
  - What are the implications concerning rotations or size/scale changes of your images?
  - Can you reduce the complexity of your task by providing other training and test data?

If you have the time, try to realize some of the suggested improvements.

[ ]: