momen

by asa asd

Submission date: 24-Feb-2023 02:41PM (UTC-0500)
Submission ID: 2022251646
File name: gy_toolkit_for_animal_pose_estimation,_tracking_and_analysis.pdf (4.39M)
Word count: 4994
Character count: 28567

October University for Modern Sciences and Art

Faculty of Computer Science

Graduation Project

A Pharmacology toolkit for animal pose estimation, tracking and analysis

Supervisor: Dr. Ayman Ezzat, and Dr. Yasmine Farouk

Name: Dema Mohamed Ibrahim Saleh

ID: 200237

Name: Moemen Ahmed Mohamed

ID: 201813

February 24, 2023

Abstract

Since the advent of device camera technology and the creation of deep networks, pose estimates have significantly improved in the field of computer vision. Animal posture studies are included in the study on posture estimates since anticipating animal poses is a vital initial step in understanding animal behaviour. In an open field experiment, the crossing, rearing, and grooming behaviours were identified using two distinct investigations. Although the second focused on grooming and raising, the first was more concerned with crossing. Both experiments were successful in counting the behaviours after utilising different photo processing algorithms. The objective outcome of identifying and monitoring various mouse behaviours including crossing and raising in an open-field experiment is accomplished through the use of machine vision, deep learning networks, euclidean distance comparison techniques, and picture processing.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

The best model for posture estimation uses video footage along with transparent reflecting markers attached to interesting locations. It greatly facilitates post-mortem study and allows precise monitoring of bodily parts within the area of motor control. Such devices, meanwhile, can be expensive and annoy animals. The trackable characteristics must also be established before recording, which predetermines the markers. This decreases video data's low degree of intrusiveness, which is one of its benefits. A substitute for employing physical markers is to fit skeleton- or contour-based models. Remarkably, even animals with stationary heads can experience oscillations that are typically referred to as "noise" in tests by trials, despite the fact that certain uninstructed motions might happen evocatively and others may be linked to trial occurrences. It is crucial to thoroughly monitor animal movements, especially those of certain body parts, and link them to cerebral activity in order to solve this issue. Examples on experiments conducted are the object recognition experemnet as shown in Figure 1,where a mouse is inside a closed off area and is introduced to two objects, and the open field experiment as shown on Figure 2.

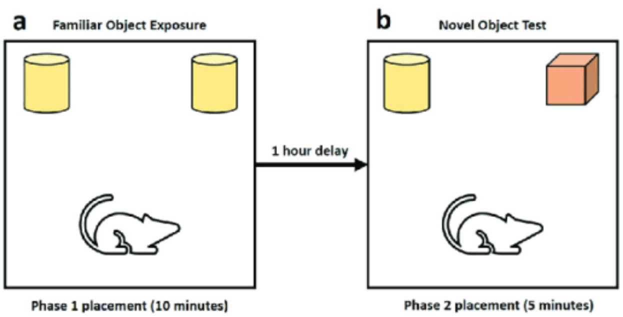

Figure 1.1: Object recognition experemnt

Figure 1.2: Open field

Pharmacy students invest a substantial amount of time in comprehending the behavioral patterns of rats through a diverse range of testing scenarios, given their common utilization in pharmaceutical research for assessing drug effectiveness and safety prior to human use. An invaluable tool for these students is the software application Anymaze, which enables the recording and analysis of animal behaviours, such as rats, across a broad spectrum of experimental settings. However, the considerable cost of Anymaze may render it financially unfeasible for some students. Nevertheless, owing to the software's extensive capabilities and utility in their work, numerous students and researchers opt to procure it despite the expense.

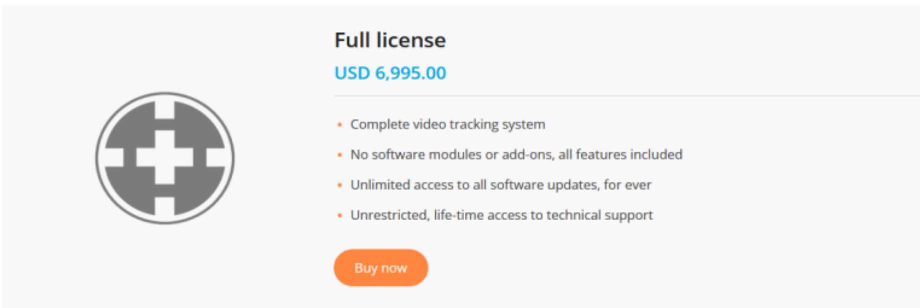

Figure 1.3: Anymaze toolkit

Pose estimation is used to detect the body's pose and use it as a reference for prediction. There are multiple applications that use pose estimation in animal tracking. In automation of labor, it can be used to using movement analysis methods .It can also be utilized to identify the type or behavior of an animal by detecting the key points of an animal's skeleton.

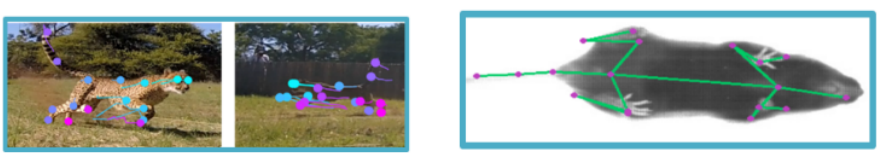

Figure 1.4: Pose Estimation on a cheetah Bottom up view on pose estimation of Mouse

1.2 Problem statement

The development of image processing technology has led to the creation of several unique methods for studying rat behaviour. Due to their sensitivity to texture and colour features and consequent dependence on the background environment, these approaches are unreliable for long-term monitoring tasks. Interestingly, mice's bodies often hide their limbs in everyday circumstances, which makes exact annotations a difficult problem. Four creatively defined point locations—the mouth, the left ear, the right ear, and the tail root—are employed to address this problem.

1.3 Objective

The objective is to precisely and consistently extract the mouse's critical skeleton points, which will subsequently be utilised to effectively track, analyse, and categorise certain posture behaviours that call for specific tracking patterns. Tasks that require a lot of labour and time will benefit from automation. Moreover, techniques for movement analysis have been proposed to quantify the relevant activities.

1.4 Motivation

Before, collected data was processed manually, which is a labor-intensive, time-consuming, and error-prone technique that is unacceptable impracticable given the tremendous rates of data gathering that are currently in place. Conversely, advances in computer vision have consistently affected data processing methods that need less human labour. Since the introduction of sensor camera technology and the development of deep networks, pose estimates have grown significantly in the field of computer vision. The research on stance estimations, which goes beyond studies of hands and humans because evaluating animal poses is a key step in understanding animal behaviour, includes studies of animal position.

1.5 Thesis layout

The first chapter: This chapter sets the context and background for the research, defines the research problem or question, and outlines the objectives and scope of the study. It should also include a brief review of the relevant literature and research that has been done on the topic. The second chapter: This chapter provides a more detailed and comprehensive overview of the existing literature and research on the topic, including theories, concepts, and empirical findings. It identifies the gaps, inconsistencies, and controversies in the literature and explains how the research study aims to address these issues. The third chapter: This chapter outlines the research design, methods, and procedures used in the study. It explains how the data was collected, analyzed, and interpreted, and discusses the strengths and limitations of the research approach. It should also provide a rationale for the choice of research methods and the ethical considerations involved in the study.

Chapter 2

Background and Literature Review

2.1 Background

2.1.1 Machine learning

Machine learning is a field of artificial intelligence that deals with the development of algorithms and models that can learn from data and make predictions or decisions without being explicitly programmed to perform a specific task. There are three main types of machine learning: Supervised learning: In this type of machine learning, the algorithm is trained on labelled data, meaning that the data is already categorised, and the algorithm must learn to make predictions based on this information. Unsupervised learning: In unsupervised learning, the algorithm is given a dataset without any labels and must find patterns and relationships within the data on its own. Reinforcement learning: In reinforcement learning, the algorithm learns through trial and error, receiving rewards for making the correct decisions and penalties for making the wrong ones. In all of these cases, the goal of machine learning is to enable the algorithm to generalise from the data it has been trained on to new, unseen data. This is what allows machine learning algorithms to make predictions or decisions that are useful in real-world applications.

2.1.2 Neural Network and Deep Learning

Neural networks and deep learning are subfields of machine learning that are inspired by the structure and function of the human brain. A neural network is a type of machine learning algorithm that is made up of interconnected nodes, which are modelled after neurons in the human brain. These nodes are organised into layers, with the input layer receiving data, one or more hidden layers processing the data, and the output layer producing predictions or decisions. Deep learning refers to neural networks with multiple hidden layers, which allows them to learn and model complex relationships in the data. The additional layers enable deep learning algorithms to automatically extract and learn hierarchical representations of the data, such as low-level features in the early layers and high-level abstractions in the deeper layers. Deep learning has been especially successful in tasks such as image classification, speech recognition, and natural language processing. It has also been used to achieve state-of-the-art results in many other domains, such as game playing, drug discovery, and financial forecasting. In summary, neural networks and deep learning are powerful machine learning techniques that have enabled significant advancements in artificial intelligence and have a wide range of real-world applications.

2.2 Previous Work

2.2.1 Markerless Rat Behavior Quantification With CNN

Without the use of invasive sensors or markers, Jin T. et al. focused on assessing joint mobility and determining rat landmark sites. Both the four-stage convolution cascade network (CCN) and the two-stage cascade hourglass network (CHN) are designed to extract features. Three distinct approaches—fully connected regression (FCR), heatmap maximum position (HMP), and heatmap integral regression (HIR)—are used to determine the positions of the landmark points. Jin also recommends a normalised assessment standard to evaluate these diverse network designs and co-ordinate computation techniques. It is demonstrated that while the CCN structure produces more accurate results, the CHN structure requires less processing. After thoroughly examining different network topologies and coordinate calculation methodologies, it is shown that the CCN+HIR approach is the most effective for small-sized feature-maps and that it can also be used when the feature-map size is increased. [4]

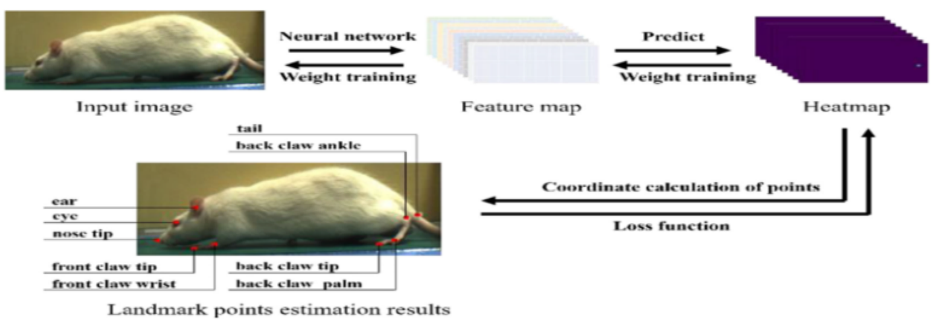

Figure 2.1: Shows steps of CNN

2.2.2 FreiPose: A Framework for Animal Motion Capture

Christian et al. present FreiPose, a deep learning-based motion capture tool that reconstructs the posture of freely moving animals without the need of markers. Instead of triangulating 2D pose reconstructions, FreiPose directly reconstructs body poses and motion trajectories in 3D, resulting in extraordinary accuracy. They conducted a comprehensive examination of the problem and combined information from all angles into a single 3D reconstruction, surpassing the state-of-the-art in terms of the median error in freely moving mice by 49.4They show that FreiPose replicates past studies on the brain correlates of body orientations and also permits spontaneous behaviour categorization, where intuitively important groups and associated variables naturally arise from [1].

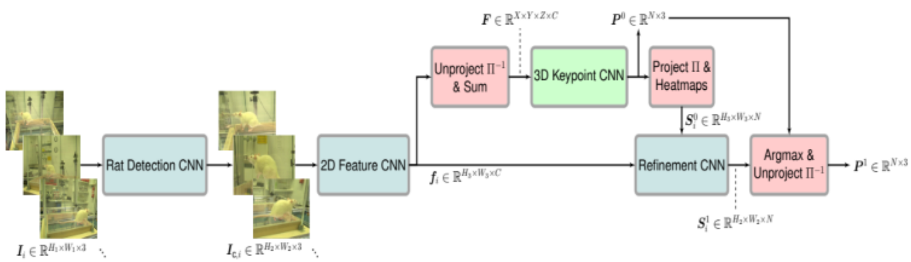

Figure 2.2: Shows steps from 2D CNN to 3D CNN

2.2.3 Multi-animal DeepLabCut

Lauer et al. (2022) developed both bottom-up and top-down approaches for multi-animal DeepLabCut (maDLC), which offer users a broader range of methods to choose from. Although the study shows that the bottom-up pipeline is typically used, the top-down techniques might be more useful for certain applications. Both methods have their own limitations and features. The authors aimed to create adaptable and high-performance code with minimal human input, allowing for 3D posture estimation of multiple animals using the code. They incorporated this multi-animal version into DeepLabCut-Live, their real-time software. During the tracking stage, users can specify the number of animals identified in a given video, and they were able to track up to 14 animals in their study. However, the camera resolution is a limiting factor, as it is necessary to locate an animal's key points. Consequently, different tracking devices may be more suitable for smaller animals. All of the steps, from posing to tracking, can be completed in ten lines of code or entirely through a GUI, without requiring any programming [3].

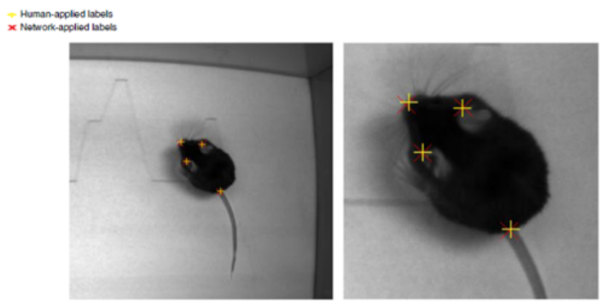

Figure 2.3: Detect body parts

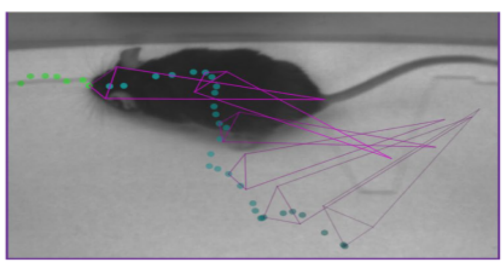

Figure 2.4: Detect body movement

2.2.4 Mouse Action Recognition System

The software pipeline for the Mouse Action Recognition System (MARS), a tool for automated study of social behaviours in mice, is described in the article. The pipeline uses deep neural networks to recognise and categorise mouse activity from video recordings, including grooming, fighting, and sniffing. By contrasting the findings of the pipeline with manual annotations of the same behaviours, the authors verify the accuracy and effectiveness of the system. By examining the impact of genetic and pharmacological changes on mouse social behaviours, they also highlight the usefulness of the technique. In conclusion, the MARS pipeline offers a potent tool for high-throughput investigation of mouse social behaviours, which can help in unravelling the brain processes underlying these behaviours [2].

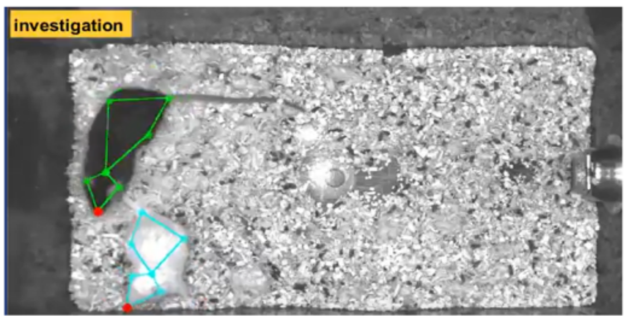

Figure 2.5: Investigation detection

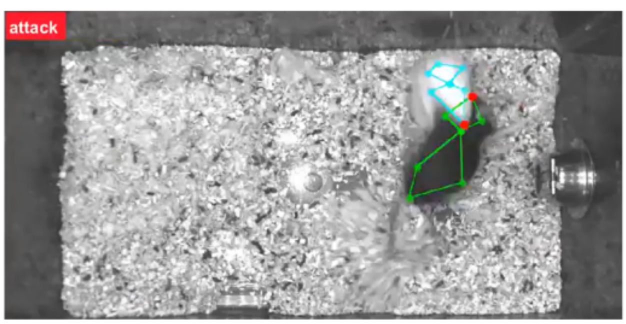

Figure 2.6: Attack detection

Chapter 3

Material and Methods

3.1 Materials

3.1.1 Data

Videos of an open field experiment provide the input utilised to assess the mouse position and trajectory. The Open Field Test is a behavioural test that is frequently used in neuroscience studies to examine the spontaneous locomotor and exploratory behaviour of rodents, usually black mice or rats. The experiment is carried out in a white open space often a square with opaque walls sectioned into a 4x4 grid, and the animal's behaviour is observed and documented via video. In the open field test, a rat is put in the centre of the arena and given a specific length of time—usually 3 to 5 minutes—to investigate the surroundings. The size of the arena can vary, but it is typically 40-50 cm wide and deep, and 30-40 cm high. The following behaviors are typically scored: Locomotion or crossing: the total distance traveled by the mouse. Center time: the amount of time spent in the center of the arena. Rearing: the number of times the mouse stands on its hind legs. Grooming: the number of times the mouse grooms itself.

Figure 3.1: Pictures from data set

3.1.2 Tools

- Python: a popular, high-level programming language that was first released in 1991 by Guido van Rossum. It is an interpreted language, meaning that it does not need to be compiled before running, but instead, it is executed line by line.

  Python is known for its readability, simplicity, and flexibility, making it an ideal language for beginners as well as experienced programmers. It has a large and active community, with a wealth of resources available to learn and use the language.

  Python has a wide range of applications, including web development, data analysis, artificial intelligence, scientific computing, automation, and more. It has a vast standard library, which includes modules for various purposes, such as working with files, networking, and databases.

  One of the most significant advantages of Python is its versatility and ease of use. It has an intuitive syntax and supports many programming paradigms, such as object-oriented, procedural, and functional programming.

- Anaconda: Anaconda is a popular distribution platform and environment for data science and scientific computing in Python. It includes a package manager, an environment manager, and a collection of over 1,500 open-source packages, including the popular NumPy, Pandas, Matplotlib, and Jupyter Notebook.

  The Anaconda distribution simplifies the process of setting up and managing data science environments by providing a comprehensive package of tools and libraries. It also provides a graphical user interface, called Anaconda Navigator, that allows users to manage and launch applications.

  One of the key features of Anaconda is its environment management system, called conda. Conda allows users to create, manage, and switch between multiple environments, each with its own set of packages and dependencies. This makes it easier to manage different projects with varying requirements without having to worry about conflicts between packages.

  Anaconda is available for Windows, macOS, and Linux, and it provides a consistent environment across all platforms. It is widely used in industry, academia, and research, and it is supported by a large and active community of developers.

  Overall, Anaconda is a valuable tool for data scientists and scientific computing, providing a user-friendly interface for managing environments, packages, and applications.

- OpenCV (Open Source Computer Vision Library): a popular open-source computer vision and machine learning software library that was first released in 2000. It is written in C++ and has interfaces for various programming languages, including Python, Java, and MATLAB.

  OpenCV is designed to provide developers with a comprehensive set of tools for building computer vision applications, including image and video processing, object detection and recognition, face detection and recognition, machine learning, and more.

- Convolutional neural networks (CNNs): a type of artificial neural network that are particularly well-suited for processing and analysing images and other types of multidimensional data. They are widely used in computer vision applications, such as image and video recognition, classification, segmentation, and object detection. CNNs are inspired by the structure and function of the visual cortex in animals, which consists of multiple layers of neurons that are tuned to different features in the visual field. Similarly, a CNN consists of multiple layers of artificial neurons that learn to recognize increasingly complex features in the input data, such as edges, corners, and textures. The key building block of a CNN is the convolutional layer, which applies a set of learnable filters or kernels to the input data to extract local features. The output of each filter is a feature map that represents the presence or absence of the filter's specific pattern at each location in the input data. The filters are trained using backpropagation and gradient descent to optimise a loss function, such as cross-entropy or mean squared error. CNNs typically include other types of layers as well, such as pooling layers that downsample the feature maps, activation layers that apply a nonlinear activation function, and fully connected layers that perform a classification or regression task based on the extracted features. The layers are arranged in a hierarchical fashion, with earlier layers learning simple features and later layers learning more abstract and complex features. The advantages of CNNs include their ability to learn and recognize complex patterns and features in high-dimensional data, their ability to handle translation and distortion invariance, and their ability to scale to large datasets and deep architectures. CNNs have achieved state-of-the-art performance on a wide range of computer vision tasks, including image classification, object detection, semantic segmentation, and video analysis.
- DeepLabCut: an open-source software toolkit for markerless pose estimation of animals and humans. The software uses deep learning algorithms to analyse videos of moving animals or humans and automatically track the movement of body parts or features, such as joints, paws, or whiskers. The name DeepLabCut comes from the combination of deep learning, labelling, and tracking.The software is based on a convolutional neural network (CNN) architecture and requires users to provide annotated data in the form of labelled images or videos, which are used to train the network to recognize and track specific features or body parts. Once the network is trained, DeepLabCut can be used to automatically track the movement of the labelled features in new videos. This enables researchers to study animal or human behaviour in a non-invasive manner, without the need for invasive techniques such as implanting markers or electrodes. The software also includes tools for data visualisation, quality control, and statistical analysis. DeepLabCut is a powerful tool for analysing complex movements and behaviours, and it has been used in a wide range of research fields, including neuroscience, biology, psychology, sports science, and computer vision. The software is freely available on GitHub, and it has a large and active user community that provides support, tutorials, and examples of its use in different applications.
- FreiPose: an open-source software package for 2D pose estimation of humans in images and

videos. The software is based on the state-of-the-art deep learning architecture, and it can accurately estimate the positions of body joints, such as elbows, shoulders, and knees, in real-time.The software is based on a deep convolutional neural network (CNN) architecture that has been trained on a large dataset of human pose annotations. This training data includes thousands of images of human subjects in different poses and under different lighting conditions, which enables the model to generalise well to new images and videos. FreiPose is designed to be highly accurate and robust, even in challenging environments such as low light or occluded poses. The software also includes advanced features such as multi-person pose estimation, 3D pose estimation, and real-time video processing.

3.1.3 Environment

- CPU — Intel(R) Core(TM) i7-9750H CPU @ 2.60GHz

- RAM — 16 GB

- GPU - NIVIDIA GTX1660 Ti

- Operating System — Windows

PyCharm is an integrated development environment (IDE) for the Python programming language. It is developed by JetBrains and is available in two versions - Community and Professional. The Community version is free and open-source, while the Professional version is a paid version with more advanced features. PyCharm provides a complete set of tools for developing Python applications. It includes a code editor with advanced features like code completion, code analysis, debugging, and testing. The IDE also comes with features like project management, version control integration, and support for multiple frameworks and libraries. Some of the key features of PyCharm include:

Intelligent Code Editor: PyCharm's code editor provides intelligent features like auto-completion, syntax highlighting, and error highlighting to make coding faster and more efficient.

Debugging and Testing: PyCharm includes a powerful debugger that allows developers to step through code and identify issues. It also has built-in support for testing frameworks like unittest, pytest, and doctest.

Project Management: PyCharm provides project management tools that allow developers to organise their code and assets in a single location.

Version Control Integration: PyCharm supports popular version control systems like Git, SVN, Mercurial, and Perforce, making it easy to collaborate with other developers. Plugin Ecosystem: PyCharm has a large plugin ecosystem that allows developers to extend its functionality with additional features and integrations. Overall, PyCharm is a comprehensive IDE that helps developers to write, debug, and test Python applications more efficiently.

3.10

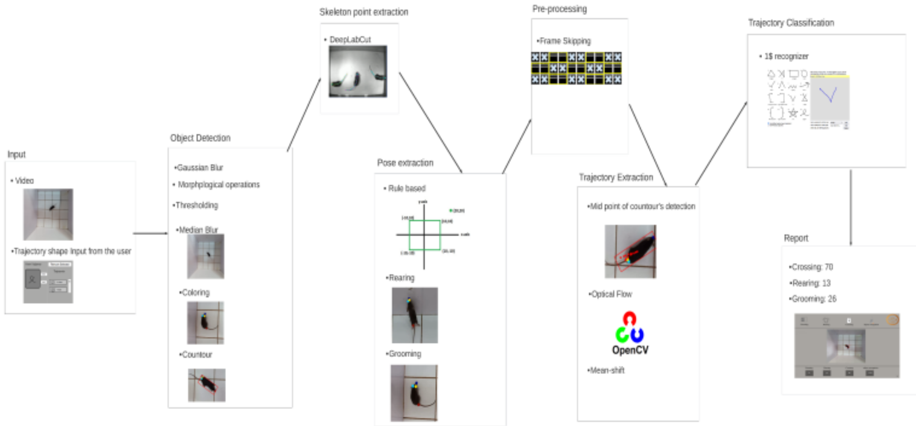

Figure 3.2: System architecture

3.2 Methods

3.2.1 System architecture Overview

The system follows a set of 8 phases in order to process data and generate reports. These phases are as follows:

1. Input:

Figure 3.3: Input

The inputs for the application consist of two components: a video with a duration of 3 to 5 minutes, and a trajectory shape input provided by the user.

2. Object detection:

Figure 3.4: Object detection

In the second phase of the project is image processing pipeline a mix of many methods is used, such as Gaussian Blur, Morphological operations, Thresholding, and Contour detection, to remove noise from the video frame samples.

Gaussian Blur is a digital image processing technique used to smooth out an image and reduce noise.

Morphological operations are a set of digital image processing techniques used to analyse and manipulate the geometric structure of an image. These operations are based on the shape and size of the objects in an image, and they can be used to perform a variety of tasks, such as noise removal, edge detection, image segmentation, and feature extraction.

Thresholding is a simple but powerful technique in digital image processing used to separate objects or regions of interest from the background in an image. It involves converting a grayscale or colour image into a binary image, where pixels are either assigned a value of 0 or 1 based on a predetermined threshold value.

Contour detection is a fundamental technique in computer vision used to identify the boundaries of objects in an image. It involves creating a curve that connects all the continuous points along the boundary of an object that have the same colour or intensity value. The technique can be used to detect objects of a certain colour chosen by the user, and it can also be used to extract shape information from an image. The resulting contours can be used to locate the objects in an image, extract features of interest, and perform tasks such as image segmentation and object recognition. Contour detection is performed using various image processing techniques, such as edge detection, smoothing, and thresholding. It is widely used in many applications, including robotics, medical imaging, security systems, and industrial inspection, among others.

3. Skeleton point extraction:

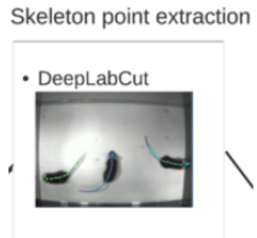

Figure 3.5: Skeleton point extraction

In the third phase of the image processing pipeline, DeepLabCut is utilised, a software tool that leverages deep learning to track the movement of animals and other objects in videos. This phase involves using DeepLabCut's markerless pose estimation capabilities to precisely locate and track specific body parts or landmarks in the video frames. DeepLabCut is a software tool that utilises deep learning to track the movement of animals and other objects in videos. Its primary purpose is to provide markerless pose estimation in complex environments with high accuracy and speed. It has gained popularity across a wide range of research fields, such as neuroscience and robotics, for its ease of use, customizable training pipelines, and open-source nature. DeepLabCut has a user-friendly interface that simplifies the process of analysing videos and tracking specific body parts or landmarks. The software leverages a neural network-based approach to learn the unique features of objects in each frame of the video, allowing it to accurately identify their positions. The system's versatility and adaptability make it ideal for use in numerous applications, including behavioural biology and biomechanics. Overall, DeepLabCut offers a powerful and practical tool for analysing videos and extracting valuable insights from complex environments. Its straightforward design and flexible architecture make it accessible to researchers and practitioners across a broad range of study fields.

4. Pose extraction:

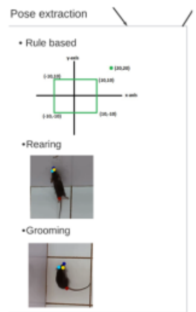

Figure 3.6: Pose extraction

Pose extraction is a technique used to determine the behaviour of a rat, such as rearing or grooming, by a rule-based model. This model is built using mathematical operations that compare the (x,y) points of the rat's features, including its nose, two eyes, and tail, to determine whether they are in a normal or abnormal position. If the rat's features are abnormal, the model evaluates whether the rat is stretching or compressing. If the rat is stretching, it is exhibiting the behaviour of rearing, while if it is compressing, it is grooming. This approach utilises rule-based algorithms to classify the behaviour of the rat based on specific criteria, allowing for a systematic and standardised method of observation and analysis.

5. Pre-processing:

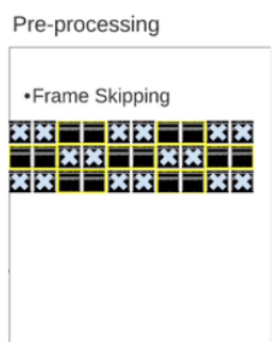

Figure 3.7: pre-processing

Pre-processing techniques are implemented to prevent errors such as frame skipping.

6. Trajectory Extraction:

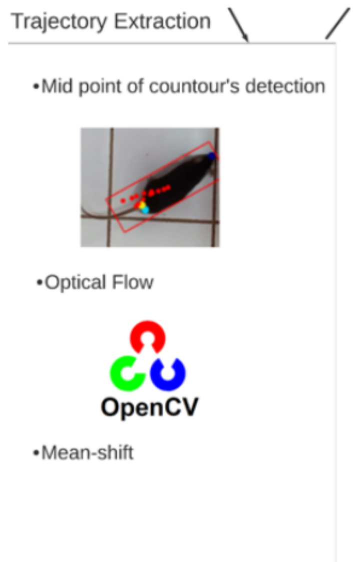

Figure 3.8: Trajectory Extraction

The rat tracking system utilizes a multi-stage approach to accurately track the rat's movements. Firstly, the system detects the rat's position by calculating the midpoint of the rat's contour using contour detection algorithms. This allows the system to track the rat's position while it is moving, and the system saves the points of the rat's trajectory as it moves. To reduce tracking errors and improve accuracy, the system applies the mean-shift technique every 30 points. This technique adjusts the tracking window based on the movement of the rat, allowing the system to track the rat more accurately and compensate for any changes in the rat's size or shape.The rat's destination is detected using various techniques to count the number of crossings. In addition to using contour detection and mean-shift techniques, the system also employs optical flow estimation. This technique estimates where points could be in the following image sequence and calculates the velocity of locations inside the images. This helps to further refine the rat's trajectory and provide more accurate tracking data. By combining these techniques, the rat tracking system is able to accurately and reliably track the rat's movements and provide valuable data for behavioral analysis.

7. Trajectory classification:

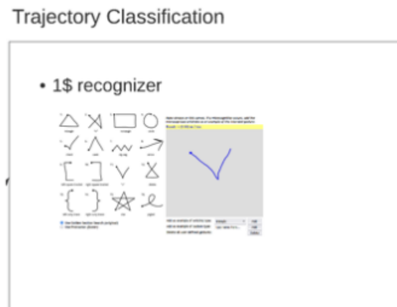

Figure 3.9: Trajectory classification

The trajectory shape of the rat is then compared to the trajectory shape that the user input, using algorithms such as the 1 Dollar recognizer.

8. Report:

Figure 3.10: Report

Finally, algorithms are employed to provide a report on an application on the rat's behaviour, including the number of crossings, rearing, and grooming.

Bibliography

[1] Christian Zimmermann, Artur Schneider, Mansour Alyahyay, Thomas Brox, Ilka Diester. FreiPose: A Deep Learning Framework for Precise Animal Motion Capture in 3D Spaces. bioRxiv 2020.02.27.967620; doi: https://doi.org/10.1101/2020.02.27.967620

[2] Cristina Segalin, Jalani Williams, Tomomi Karigo, May Hui, Moriel Zelikowsky, Jennifer J Sun, Pietro Perona, David J Anderson, Ann Kennedy (2021) The Mouse Action Recognition System (MARS) software pipeline for automated analysis of social behaviors in mice eLife 10:e63720 https://doi.org/10.7554/eLife.63720

[3] Lauer, J., Zhou, M., Ye, S. et al. Multi-animal pose estimation, identification and tracking with DeepLabCut. Nat Methods 19, 496–504 (2022). https://doi.org/10.1038/s41592-022-01443-0

[4] Tianlei, J., Feng, D., Zhenyu, Y., Shifan, Y., Xuyi, C., Yu, L., Qingyu, Y., Fengzeng, J. (2020). Markerless Rat Behavior Quantification With Cascade Neural Network. Neurorobot. 10.3389/fnbot.2020.570313

momen

ORIGINALITY REPORT

19%	13%	6%	11%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to October University for Modern Sciences and Arts (MSA) Student Paper	2%
2	doctorpenguin.com Internet Source	1%
3	Submitted to University of Greenwich Student Paper	1%
4	www.ncbi.nlm.nih.gov Internet Source	1%
5	Submitted to University of Southern Queensland Student Paper	1%
6	Submitted to Higher Education Commission Pakistan Student Paper	1%
7	www.nature.com Internet Source	1%
8	Submitted to University of Hertfordshire Student Paper	<1%

9	Submitted to University of Bedfordshire Student Paper	<1%
10	www.coursehero.com Internet Source	<1%
11	Submitted to Dr. S. P. Mukherjee International Institute of Information Technology (IIIT-NR) Student Paper	<1%
12	Submitted to New York College in Athens, Greece Student Paper	<1%
13	www.ukessays.com Internet Source	<1%
14	Submitted to Middle East College of Information Technology Student Paper	<1%
15	Submitted to National School of Business Management NSBM, Sri Lanka Student Paper	<1%
16	Christian Zimmermann, Artur Schneider, Mansour Alyahyay, Thomas Brox, Ilka Diester. "FreiPose: A Deep Learning Framework for Precise Animal Motion Capture in 3D Spaces", Cold Spring Harbor Laboratory, 2020 Publication	<1%
17	www.scielo.br Internet Source	<1%

18	Submitted to Cheshire College South and Wes Student Paper	<1%
19	eprints.nottingham.ac.uk Internet Source	<1%
20	infomall.org Internet Source	<1%
21	www.biorxiv.org Internet Source	<1%
22	Submitted to Edith Cowan University Student Paper	<1%
23	www.politesi.polimi.it Internet Source	<1%
24	www.researchgate.net Internet Source	<1%
25	Submitted to Tower Hamlets College Student Paper	<1%
26	Submitted to University of Newcastle Student Paper	<1%
27	Wael A. Altabey, Sallam A. Kouritem, Mohammed I. Abouheaf, Nabil Nahas. "Research in Image Processing for Pipeline Crack Detection Applications", 2022 International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME), 2022	<1%

Publication

28	github.com Internet Source	<1 %
29	intellipaat.com Internet Source	<1 %
30	ml4a.github.io Internet Source	<1 %
31	Jiss Kuruvilla, Dhanya Sukumaran, Anjali Sankar, Siji P Joy. "A review on image processing and image segmentation", 2016 International Conference on Data Mining and Advanced Computing (SAPIENCE), 2016 Publication	<1 %
32	dspace.mit.edu Internet Source	<1 %
33	Submitted to Brown University Student Paper	<1 %
34	amslaurea.unibo.it Internet Source	<1 %
35	www.da6nci.com Internet Source	<1 %
36	Ghosal, S.. "Orthogonal moment operators for subpixel edge detection", Pattern Recognition, 199302 Publication	<1 %

37	en.cs.uni-paderborn.de Internet Source	<1%
38	www.i2tutorials.com Internet Source	<1%
39	ssg.mit.edu Internet Source	<1%
40	dashboard.hpc.unimelb.edu.au Internet Source	<1%
41	web.archive.org Internet Source	<1%
42	www.mkcr.cz Internet Source	<1%
43	www.techgeekbuzz.com Internet Source	<1%
44	Jessy Lauer, Mu Zhou, Shaokai Ye, William Menegas et al. "Multi-animal pose estimation, identification and tracking with DeepLabCut", Nature Methods, 2022 Publication	<1%
45	Artur Schneider, Christian Zimmermann, Mansour Alyahyay, Florian Steenbergen, Thomas Brox, Ilka Diester. "3D pose estimation enables virtual head-fixation in freely moving rats", Cold Spring Harbor Laboratory, 2022 Publication	<1%