Faculdade de Engenharia da Universidade do Porto

FEUP

Automatic Behavior Recognition in Laboratory Animals using Kinect

João Pedro da Silva Monteiro

Master in Bioengineering

Supervisor: Jaime dos Santos Cardoso (PhD)
Co-Supervisor: Paulo de Castro Aguiar (PhD)

July, 2012

Automatic Behavior Recognition in Laboratory Animals using Kinect

João Pedro da Silva Monteiro

Master in Bioengineering

Approved in oral examination by the committee:
Chair: Artur Cardoso (PhD)
External Examiner: Paulo Miguel de Jesus Dias (PhD)
Supervisor: Jaime dos Santos Cardoso (PhD)
Co-Supervisor: Paulo de Castro Aguiar (PhD)

23th July, 2012

Abstract

Animal behavior assessment plays an important role in basic and clinical neuroscience. Although assessing the higher functional level of the nervous system is already possible, behavioral tests are extremely complex to design and analyze. Animal's responses are often evaluated and quantified manually by direct observation of the animal or indirectly by analysis of video captured for this purpose. This manual evaluation is subjective, extremely time consuming, poorly reproducible and potentially fallible.

In order to test alternative solutions to this problem, a project was then proposed to evaluate the use of the recently available consumer depth cameras, such as the Microsoft Kinect, for characterization and detection of simple behavioral patterns of laboratory animals. The hypothesis is that the simultaneous capture of video and depth information, provided by systems like Kinect, should enable a more feasible and robust method for automatic behavioral recognition and tracking of the position of laboratory animals, over previously available systems.

To fulfill the main objective of this project a general framework was considered that may be briefly described as follows: features are computed to convert an input video sequence composed of depth and color images, into a representation which is used for the automated recognition of the animal’s behavior by a statistical supervised classifier.

Seeking accuracy and overall reduction of resource consumption, some work was done on image segmentation. Four different approaches were tested and evaluated resulting from such effort a tracking procedure for laboratory mouse using depthmap information.

Subsequently to being able to robustly extract masks surrounding mice against involving arena we tested an observation model to infer the center of mass and the other parameters of an ellipse that approximates the mice contour. Each parameter contribution to recognition of behaviours of interest is evaluated and is proposed the extent of use of spatial temporal features in different time windows.

Throughout the present work we established a potential method for recognition of behavior for a singly housed mouse in an open-field apparatus capable of identifying walking, resting, rearing and micro-movement occurrences.

Resumo

A avaliação do comportamento animal desempenha um papel importante na neurociência básica e clínica. Embora avaliar o nível funcional mais elevado do sistema nervoso seja já possível, testes comportamentais são extremamente complexos de conceber e analisar. As respostas dos animais são frequentemente avaliadas e quantificadas manualmente por observação direta do animal ou indiretamente por meio de análise de vídeo capturado para este fim. Esta avaliação manual é, portanto, subjetiva, extremamente morosa, pouco reprodutível e potencialmente falível.

Tendo como propósito a procura de alternativas para este problema, um projeto foi então proposto para avaliar a utilização de sensores de profundidade recentemente disponibilizados ao consumidor, tais como a câmara Kinect da Microsoft, para caracterização e deteção de padrões de comportamento de animais de laboratório. A hipótese assumida por esta dissertação é a de que captação simultânea de informação de cor e profundidade, proporcionada por sistemas como Kinect, deverá permitir um método mais viável e robusto para o reconhecimento automático e rastreio da posição dos animais de laboratório, em relação aos já sistemas disponíveis.

De forma a alcançar o principal objetivo deste projeto foi considerada uma abordagem que pode ser descrita resumidamente da seguinte forma: são determinadas características que convertam uma sequência de entrada de vídeo composto de imagens de profundidade e de cor, numa representação que permite o reconhecimento automático dos comportamentos do animal através de um classificador estatístico supervisionado.

Procurando precisão e redução global de consumo de recursos, foi elaborado algum trabalho em segmentação de imagem. Quatro diferentes abordagens foram testadas e avaliadas resultando desse mesmo esforço um método de rastreamento de ratinhos de laboratório que usa informação de profundidade.

Subsequentemente a ser capaz de extrair robustamente máscaras dos ratinhos contra a arena circundante, testámos um modelo de observação para inferir o centro de massa e outras características a partir de uma elipse que aproxima o contorno dos ratinhos. Foi avaliada a contribuição de cada característica para o reconhecimento de comportamentos de interesse e é proposto o uso de janelas temporais diferenciadas para cada uma das características de forma a estabelecer uma noção de contexto temporal.

Ao longo do trabalho foi estabelecido um método para reconhecimento do comportamento de ratinhos na situação de teste de campo aberto capaz de identificar deambulação, inatividade, exploração vertical e ainda micro-movimentos.

Acknowledgements

Prof. Doctor Jaime Cardoso, my supervisor at FEUP, provided prompt support during the project, useful recommendations and close follow up.

Prof. Doctor Paulo Aguiar, my co-supervisor at FCUP, shared me his enthusiasm on the project, gave important insights on its importance and useful recommendations to consider on its development.

Eng. Hélder Oliveira, PhD student at FEUP guided me in all phases of the project. He advised me with important research lines that truly helped on the progression of the work and gave important insights on how to solve problems.

All the people at Visual Computing and Machine Intelligence (VCMI) group for the admirable work environment.

I also acknowledge the help and support provided by Instituto de Biologia Molecular e Celular (IBMC) for allowing using their facilities and mice to generate the training video database.

To all my family, my father and my mother for the unconditional support they have shown.

To all my friends, for helping me keep my sanity whenever I was close to losing it.

"Não sou nada.

Nunca serei nada.

Não posso querer ser nada.

À parte isso, tenho em mim todos os sonhos do mundo."

Álvaro de Campos

Contents

List of Figures

List of Tables

Abbreviations

BBU	Basic Behavior Unit
Caltech	California Institute of Technology
CBCL	Center for Biological and Computational Learning
FEUP	Faculdade de Engenharia da Universidade do Porto
fps	frames per second
HMM	Hidden Markov Model
kNN	k-Nearest Neighbour
MIT	Massachusetts Institute of Technology
NTSC	National Television System Committee
PAL	Phase Alternating Line
SBM	Static background model
SVM	Support Vector Machine

Chapter 1

Introduction

The draft human genome sequence published on February 15th 2001 [1] followed by a sequencing technology explosion, proved to be a crucial milestone for the advance of biomedical research. Despite its importance, it is recognized that gene sequencing represents only an initial step in the long road still to go before there is an understanding concerning the complex relationship between genes and behavior in normal and pathological conditions [2]. In this sense, a great effort has been developed by disciplines such as neurosciences and pharmacology for which animal experimentation remains a key instrument [3]. Among animals used in research, mice can be recognized as one of the most important model [4]. They tend to be used for analysis of behavior patterns of targeted and chemically induced mutations. Such assays are often evaluated and quantified manually, what poses as time-consuming activity in research laboratories.

1.1 The problem

The demand for automated systems of mice behavior analysis in laboratory arises primarily in order to resolve problems related not only to time and cost but also to reproducibility inherent in the assessment process conducted by people. Additionally, the availability of such systems, introduces the possibility to rethink behavior tests themselves, since the typical testing time scale can be easily extended, and thus diversify the behaviors analyzed, and the context of evaluation. Automated analysis of mouse behavior constitutes a challenge due to a large number of factors. That includes the huge variability of the conditions of conducted behavior tests, or the generic problem of behavior recognition itself. Notwithstanding, the use of automated approaches has been documented [5] for monitoring and recognition of behavior of laboratory animals. However, either approaches based on sensors or video captures still have limitations, are not widely used and may still be cost prohibitive [6].

1.2 Motivation and Objectives

Advances in computer vision and machine learning have yielded robust systems for the recognition of objects, alongside with new approaches in the recognition of behavior patterns. Examples of these advances include approaches based on better understanding of the human visual system and the combination of space-time feature extraction ([7],[8],[9],[10]). It should also be noted the research using depth images for object recognition or modeling over the last few decades. Depth images are simple representations of tridimensional (3D) information presenting several advantages over bidimensional (2D) intensity images which present the potential for greater recognition accuracy ([11],[12],[13]). While earlier depth sensors were expensive and difficult to use [14], the task has recently been greatly simplified by the introduction of real-time consumer depth cameras ([15],[16],[17]) that capture per-pixel depth information along with RGB images. Since then, a considerable growth in the amount of research in the field has taken place ([18],[19],[20]).

The previously mentioned behavior analysis challenges and the recent developments in computer vision and pattern recognition leads to the main motivation for this master thesis, namely, the application of depth acquisition technologies for the characterization of behavior in laboratory mouse.

Other systems have been developed, not only addressing the problem of tracking mice but also allowing the analysis of simple animal activities such as grooming or rearing. It is however possible to find some limitation in the current available systems, such as, lack of characterization of social behavior and some fine behavioral patterns, restrictions to camera pose, housing apparatus, lighting conditions and mice color ([21],[8]).

With the use of color and depth information obtained by a single device, we strive to promote a behavior analysis system that might be used in a less constrained manner. In particular regarding lighting conditions and colors of both the mouse and the apparatus used. Thus enabling a simplified apparatus and specimen preparation and at same time consolidating the fine analysis of behavior of mice, making it faster, simpler and more robust.

1.3 Contributions

This works contributes to the study of animal behavior making use of the areas of computer vision and pattern recognition, through the application of algorithms for automated analysis of animal behavior. Results include:

- Creation of a database of mouse behaviors taken by the Kinect and its manual annotation;
- Tracking algorithm for laboratory mouse, aided by synchronous depth and color/RGB data acquisition;
- Recognition algorithm for animal behavior analysis from visual information;
- Article published in 1st PhD. Students Conference in Electrical and Computer Engineering (StudECE 2012) with the title Depth-map Images for Automatic Mice Behavior Recognition [22].

1.4 Report's Structure

The entire thesis is divided into five chapters. Apart from Introduction (Chapter 1), the following list describes the contents of the remaining chapters:

- Chapter 2: Literature Review - A review of some published work related to the thesis topic will be presented, as well as a description of some typical mouse behavioral assessment and existing systems
- Chapter 3: Automated Recognition of Mouse Behavior – The description of methods developed, in the field of image segmentation, camera calibration, and classification.
- Chapter 4: Results and Discussion – Presentation and discussion of results obtained.
- Chapter 5: Conclusions and Future Work - Presentation of main conclusions and some notes regarding future work.

Chapter 2

Literature Review

In this chapter a literature review on mouse behavioral assessment and some fundamental background concepts are presented. The information is provided having in consideration the description of project aims, so that a final solution can be presented and justified based on the conducted literature review.

2.1 Mouse Behavioral Assessment

Manipulation of expression of individual genes within the brain, have provided an extensive set of opportunities for investigating the influence of genes on behavior. On the other hand, behavioral assays, and their application to large numbers of animals, need for more advanced in order to pace in terms of throughput with the rapidly escalating use of genetic manipulations in mice [23].

Generally behavioral assays can be categorized into two classes [5]:

- Animal models of clinical disorders
- Behavioral screening tests.

Animal models of clinical disorders represent attempts to simulate symptom clusters characteristic of particular diseases. Screening tests are used to assess the impact of genetic and pharmacological manipulations on behaviors chosen to reflect particular behavioral processes of interest. For example, the forced swim test and elevated plus maze are used to screen compounds and genetic mutations for their relevance to depression or anxiety, respectively. More concretely, to assess sensorimotor performance in rodents, a number of behavioral tasks have been designed and currently in use such as acoustic startle, sensory gating, open-field exploration and rotarod.

Alternatively, some behaviors are used as bioassays to assess the activity of particular neural pathways. Examples of such behavioral bioassays include circling behavior to test function of the brain's dopaminergic systems.

2.1.1 Motor Function and Spontaneous Exploration

Several standard behavioral tests, such as the open-field and elevated plus maze, have proven their usefulness and validity for assessing both exploratory and locomotor activity [23]. The open-field apparatus is perhaps the simplest and most economical [24]. Originally introduced as a measure of behavior in rats, open-field exploration has proven to be equally successful with mice [25]. The main purpose of open-field tests is to examine mouse responses to a new and unfamiliar environment. In addition, repeated exposure or extended session length provides a method for assessing habituation to the increasingly familiar chamber environment.

There are several variations of its exact protocol, however, taking as reference the indications from [26] it's possible to indicate that it generally consists of a square arena of adequate size (e.g., 50 × 50 cm) surrounded by walls to prevent the animal from escaping as the example shown in Figure 2.1. The box itself may be composed of either wood or plastic and the floor divided into equally spaced regions by marker pen. The observer placed at a distance from the apparatus, or watching a monitor fed by a video camera positioned above the open field, records the specific behaviors using data sheets and counters over a period of time usually between 5 and 10 minutes. Typical parameters to record may include:

- number of square crossings within the specified time;
- time spent near walls;
- time spent stretching (defined by standing still or walking slowly with a low, stretched posture);
- frequency of rearing (defined by an upright posture and forelimbs off the ground)
- frequency of grooming (defined by the forelimbs or hind limbs sweeping across the face or torso).
- Defecation frequency.

Figure 2.1- Example of Open field apparatus during motor function and spontaneous exploration test situation.

An example of registration carried out during an open-field test is presented in Figure 2.2. Concerning interpretation of the Open-field tests data [27], mice that are inactive in locomotion and defecate more often are assumed to have intense anxiety and fear. Animals' tendency to spend more time in the periphery and stretching can also provide indices of higher levels of fear and anxiety, while measures of decreased anxiety are frequency of rearing and grooming.

A more formal assessment of anxiety can be made using a modified open-field apparatus. For that, the open-field is separated into a well-lit area and a dark area and the relative time and activity in these two zones is compared. Anxiolytics increase the time spent in the well-lit zone in this light–dark test.

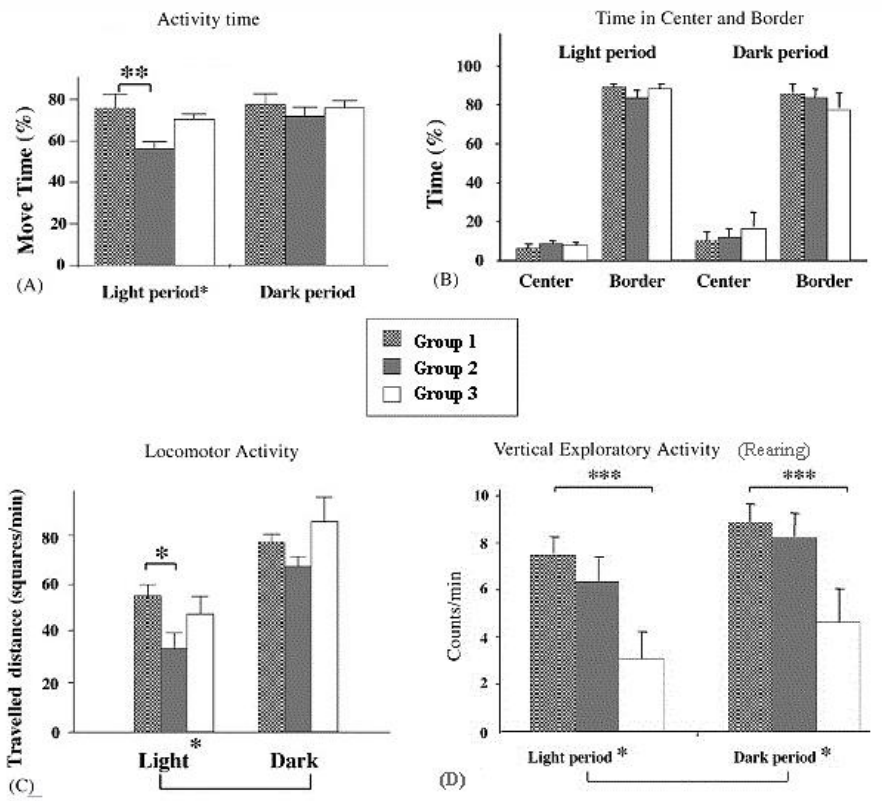

Figure 2.2 - Example of results accomplished through manual examination of the video recording of an open-field test both during the light and dark periods adapted from [25].

In addition to such approach, efforts are also underway [23] in order to achieve a comprehensive evaluation of the impact of experimental manipulations on mouse behavior reflecting the diverse and spontaneous behaviors of mice in their home-cages. Observation of behavioral patterns of animals fully acclimated to housing in cages, are been pointed out [28] as useful resources as a basis to search for genes conferring strain differences and for the selection of optimal strains for the study of particular behavioral processes.

Testing animals in their home-cage environment allows for long-term continuous observations, emphasized the relevance of studying baseline activity in the home-cage for the interpretation of behavior in novel environments. Furthermore, minimal human intervention is needed which reduces stress caused by handling and saves time-consuming human observations. By designing the home-cage environment as an automated, modular system that contains different stimuli (e.g. nesting box, light and sound stimuli, novel objects), a broad range of behaviors as a result of interacting motivational systems can be studied. It allows the distinction of novelty-induced and baseline behaviors and offers the opportunity to study circadian rhythmicity in detail [28].

2.1.2 Behavior Modeling

The study of the behavior of natural systems is a mainstay of scientific endeavors of all time, and "the general goal is to produce a description that not only explains what happens, but that can be used to predict future events" [29]. Behavior modeling can be found from natural physical systems, to live organisms' behavior study, life-like character animation, robot motion control, and automated behavior analysis from video sequences.

Hereupon, essential to the automated behavior analysis, behavior modeling embraces the tasks of feature extraction definition, as well as the creation of a generative behavior model. First, recalling the open-field scenario as an example is possible to highlight as behaviors of interest: walking, stretching, grooming, and rearing. Second, definition, characterization, and representation of these behaviors in terms of intrinsic behavioral variables, then mapped to the following factors:

- physical (spatiotemporal) features;
- the relationship between behaviors;
- the relationship between the animal and its environment.

Spatiotemporal physical features of the object that can represent and differentiate different behaviors. The features may include the object's position, posture, speed, contour or region pixels, dynamics, motion patterns, and other derived features. Features of the environment may also need to be extract. This process usually requires the ability to detect and track objects. Moreover, definition and characterization of these behaviors do not need to depend on the data acquisition system, but they can then drive the task of feature extraction from data source for basic and complex behaviors recognition, which may in turn help the interpretation of behaviors.

Third, another important component in this block is the internal generative model driving the behaviors of an animal. The behavior representation and description is inherently hierarchical. The lowest level is the spatiotemporal image sequence features. These features may be derived from the intrinsic variables that define the behavior. The next level is the basic behavior units (BBUs) defined in terms of certain spatiotemporal image features. Basic Behavior Units are the primitive actions, or activities that happen during a short period of time, which can be characterized by certain intrinsic variables or features. The basic behaviors are totally domain specific. Even in this level, it might be split to multiple levels in certain applications. Then more complex behaviors are represented as a set of BBUs with certain constraints or relationships.

2.2 Video-Based Behavior Analysis Systems

Automation of behavioral tests started with the use of electro-mechanical devices for experimental control when specific action-reaction or stimulus-response relations had to be quantified [23]. Although these systems can be used effectively to monitor the locomotor activity, fail to understand more complex and comprehensive behaviors. In this sense, the visual analysis is presented as a potentially decisive and powerful supplement, and it is this kind of approach that was attempted in the course of this thesis.

It is possible to find several published methods dealing with the problem of video-based animal behavior analysis in different ways such as:

- Labeling animal body parts and detecting behavior by extracting their movements [30];
- Identifying semi invariant features on mouse contours for behavior recognition [31];
- Applied machine learning methods to analyze rat behavior [32];
- Computation of a dictionary of about 300 space-time motion features and additional set of features derived from the instantaneous location of the animal in the cage to train a Hidden Markov Model / Support Vector Machine [21];

All these vision-based methods used standard video images (NTSC 30 fps / PAL 25 fps) and extract motion features are essentially based on from previous work for the recognition of human actions and biological motion ([30], [31], [32], [21]).

2.2.1 Mice Tracking

Solving the mice tracking problem reduces to the very challenging problem of tracking textureless, near-identical deformable objects [33]. Computer vision techniques for detection and tracking may not translate to the mice tracking problem. The mice tracking system, as any other, should consist of two major components: a mouse detection system and a tracker.

The following details the primary mice tracking challenges:

- Featureless - the interior of the mouse is nearly featureless and local features points are not reliable over many frames due to self-occlusion.
- Highly Deformable - mice deform into many shapes, sizes, and orientations. Consequently, sliding window object detection systems are not effective for detecting and tracking mice.
- Long-Term Tracking - the system must track over long-term experiments with minimal human intervention; otherwise, the system is unlikely to be adopted by the research community.
- Unpredictable Motion - mice move erratically and change directions abruptly. This complicates the use of motion models to predict future mouse locations.

Several animal vision tracking and behavior analysis systems were developed as for example the case of open source software published in [34]. Also available and used to some extent in research some commercially available and are listed here in order to establish some of the approaches of potential interest for the thesis:

- *Ethovison* [35] from Noldus Company is a comprehensive video tracking, analysis and visualization system for automatic recording of activity, movement and social interaction of various kinds of animals in an enclosure. It provides a range of features for video tracking and data analysis, and allows for automation of many behavioral tests. It uses color to distinguish animals from background and to analyze behavior patterns based on movement paths. However for behaviors as complex as grooming, the user is prompted to label it interactively [29].
- The *SMART* system from San Diego Instruments is an animal video tracking and analysis system for behavioral tests (mazes), whose analysis is mostly based on an animal's path.

- ANY-maze is a video tracking system designed to automate testing in behavioral experiments that includes statistical analysis of test results, using a conventional notebook computer and a USB or FireWire camera.
- The *Video Tracking System* from Qubit System, Inc. operates on the concept of contrast and tracks an animal's trajectory.
- The *Trackit* system from Biobserve Company tracks the animal position and orientation in 2D and for flying insects, which in turn controls the pan-tilt camera to get close-up images.
- The *Peak Motus* System from Vicon Peak Company tracks human, animal and other objects automatically with markers or based on contrast.
- The *Big Brother* system from Actimetrics Company tracks the path of the animal under study, which is the basis for further analysis.

The following table summarizes the identified tracking solutions:

Table 2.1 - Commercially available video-tracking systems

System name	Company	Website	Published references
Ethovison XT	Noldus	[36]	[35], [37]
SMART	San Diego Instruments	[38]	[39]
Any-maze	Stoelting	[40]	[41]
Video Tracking	Qubit System	[42]	n.a./n.f.
Trackit	Biobserve	[43]	[44]
Peak Motus	Vicon Peak	[45]	n.a./n.f.
Big Brother	Actimetrics	[46]	n.a./n.f.

n.a./n.f.-> not available/not found

2.2.2 Automated behavioral analysis of mouse

The majority of previous works regarding laboratory animal behavior analysis concerns largely with solutions for tracking or simpler analysis of mouse trajectory. In more recent works some of those projects started to mature into some more evolved solutions, such as commercially available system by *Noldus* [36], and dedicated commercial software *HomeCageScan*[47] by CleverSys Inc.[48].

Besides proprietary systems, worth noting two existing works on actions reccognition of solitary black mice on a white background: [8], [21]. In particular the system for monitoring the home-cage behaviour of mice proposed in [21]. Being available as open source since 2010 it comprises both feature computation and classification modules and it marked a noteworthy qualitative leap compared to previous approaches. It can be described in the following steps:

- establishment of a foreground mask for pixels belonging to the animal by means of background subtraction procedure to the input video;
- cutting a sub window centered on the animal location from each video frame;
- computation of position and velocity-based features derived from the instantaneous location of the animal in a cage, computed from a bounding box surrounding the animal in the foreground mask (10 features);
- computation of space-time motion features extracted from the a sub window derived from combinations of the response of afferent units that are tuned to

different directions of motion as found in the mammalian primary visual cortex adapted from previous work for the recognition of human actions and biological motion[10] (300 features);

- classification of every frame of a video sequence into a behavior of interest through a support vector machine classifier with hidden Markov models (SVMHMM).

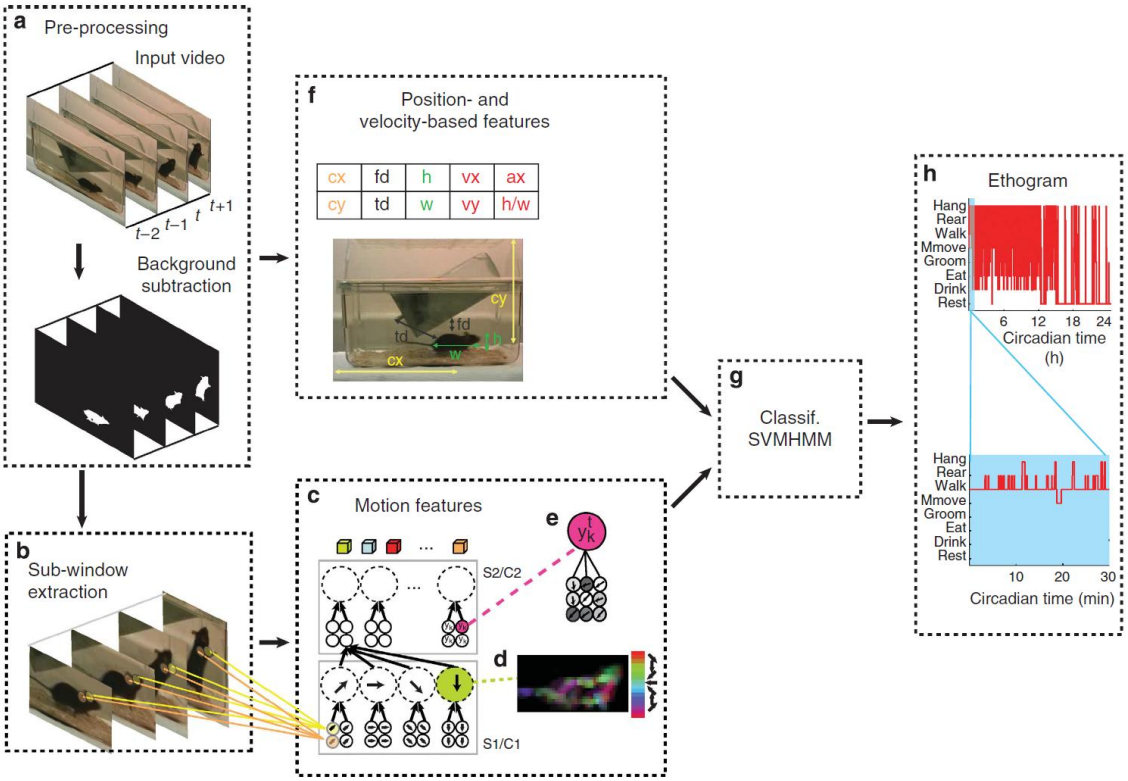

Figure 2.3 - Overview of the system for Mouse Behavior Recognition from MIT Center for Biological and Computational Learning (CBCL) for monitoring the home-cage behaviour of mice, consisting of a feature computation module (a – f) and a classification module (g). Retrieved from [21].

Having been tested against a ground truth manual annotation the system proved to outperform a specific commercial system Figure 2.4. Important to retain is also the level of agreement within annotation from different annotator that roughly exceed 70% and for the particular data set can be as low as 57% for some behavior such as groom Figure 2.5.

	System	CleverSys commercial system	Human (*'Annotator group 2'*)
'Set B' (1.6 h of video)	77.3%/76.4%	60.9%/64.0%	71.6%/75.7%
'Full database' (over 10 h of video)	78.3%/77.1%	61.0%/65.8%	

Figure 2.4 - Accuracy of the system for Mouse Behavior Recognition from MIT CBCL. Accuracies reported as averaged across frames / across behaviors (underlined numbers, computed as the average of the diagonal entities in Figure 2.5 confusion matrix; chance level is 12.5% for an eight-class classification problem).Retrieved from [21].

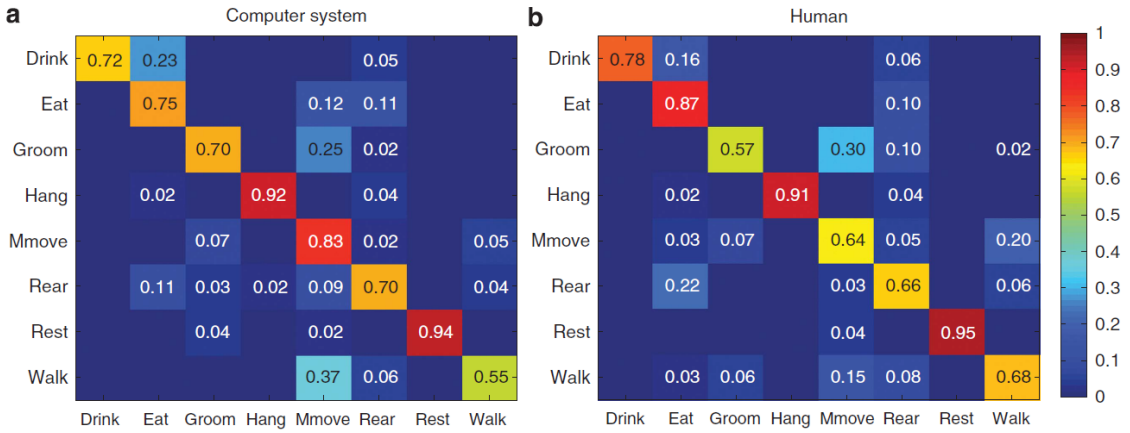

Figure 2.5 - Confusion matrices evaluated on a doubly annotated set (~1.6hrs of video) to compare the agreement between (a) the system for Mouse Behavior Recognition from MIT CBCL and human scoring, (b) human to human scoring. Adapted from [21].

More recent works continue to expand the capabilities of behavior recognition systems to be able to start characterizing social behavior and to tackle some of the constraints imposed on the analyzed scenes [33],[49].

In particular the work publish more recently [49] introduces a novel trajectory features (TF), used in a discriminative framework, explores the temporal context in behavior analysis, and studies the combination of spatiotemporal features (STF) from the side (STF side) or from the top view (STF top) both recorded at 25fps with a resolution of 640x480 pixels, 8-bit pixel depth, monochrome. The corresponding method results and confusion matrices and present in Figure 2.6 and Figure 2.7.

Features used	Without Context	With Context
TF	52.3%	58.3%
STF	29.3%	43.0%
STF Top	26.6%	39.3%
STF Side	28.2%	39.1%
(Full method) TF+STF	53.1%	**61.2%**

Figure 2.6 - Results of Social Behavior Recognition in continuous videos method from Computer Vision at Caltech. Evaluated on 237, 10min long videos were divided in two sets, 104 for training and 133 for testing. Retrieved from [49]

Figure 2.7 confirms accuracy rates between different annotators of around 70%, corroborating the statement cited above and at the same time giving a perspective of evolution mice behavior analysis systems into a state of modeling of social interactions.

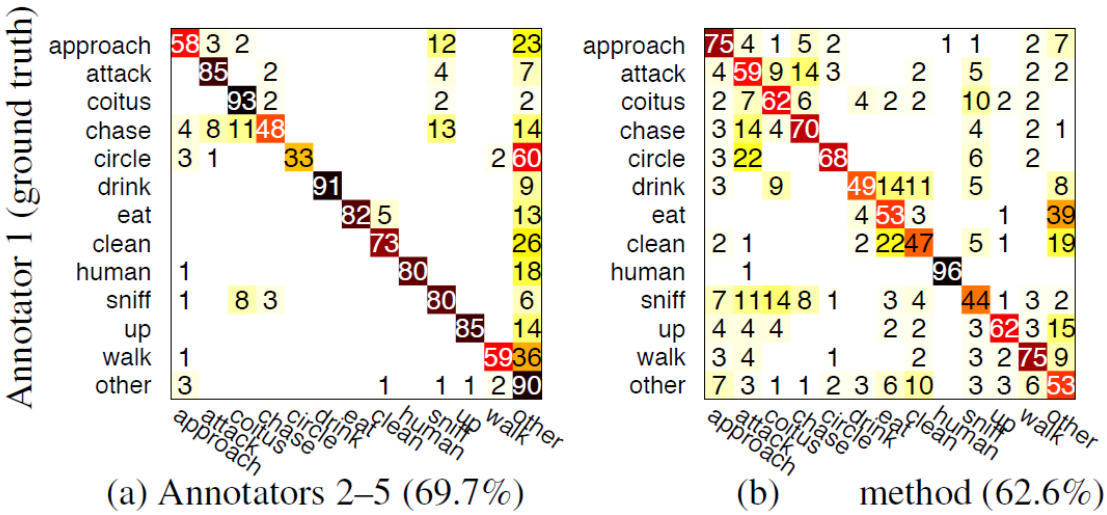

Figure 2.7 - Confusion matrices for comparison with expert's annotations on the 12 videos (~2h) that were annotated by more than two experts. (a) Comparison between 'annotator1' and a group of 4 other annotators. (b) Comparison between 'annotator1' and the output of Social Behavior Recognition in continuous videos method from Computer Vision at Caltech. Retrieved from [49]

Chapter 3

Automated Recognition of Mouse Behavior

Based on current state of the art, automated recognition of mice behavior can be done using different tools and techniques. At this project the simultaneous acquisition of depth and color information through Kinect device is suggested as a potential resource for such purpose.

On this chapter, the hypothesis and projects considered to design a video-based behavior recognition method are presented.

A generic video-based behavior recognition system may be briefly described as follows: features are computed to convert an input video sequence composed of depth and color images, into a representation which will later be used for the automated recognition of the animal's behavior by a statistical classifier. This framework (Figure 3.1) also sets the three main areas of interest preceding the implementation of the video-based mouse behavior recognition system, namely, mouse segmentation, feature extraction, and supervised learning.

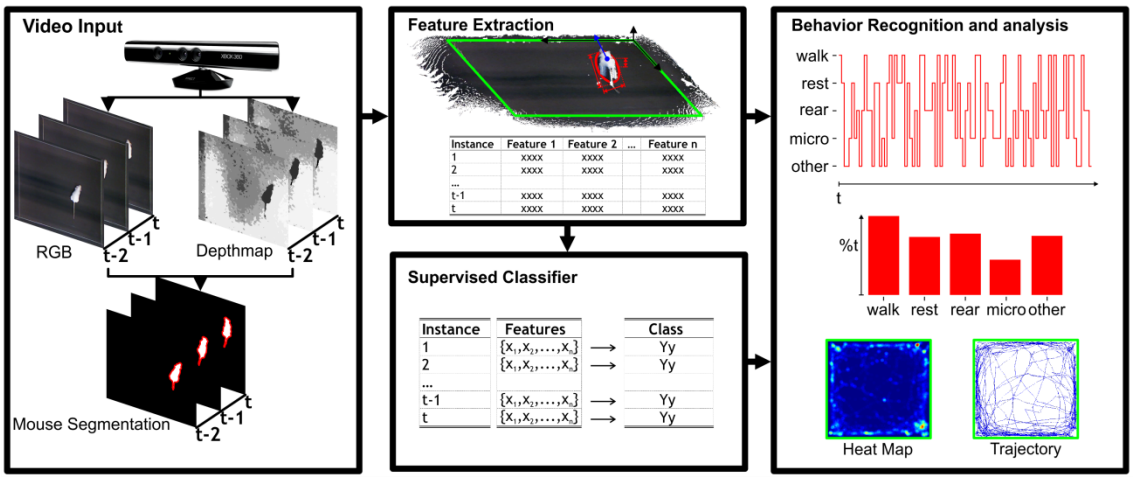

Figure 3.1 - System overview

In this chapter is also discussed the Kinect calibration method, though not explicitly present in the framework, it is necessary in order to use the matching of color and depth information. Finally, are present some notes on the database creation.

3.1 Depth and Color Camera Calibration

The Kinect sensor from Microsoft consists of a color camera rigidly attached to a depth sensor which comprises a projector-camera pair that measure per pixel disparity (more information in the appendix). Such systems have been observed to suffer from geometric distortions due to the processing performed and the inevitable tolerances in their manufacturing. Whereas a radial and tangential distortion model is usually sufficient to correct the 2D pixel positions in color cameras, depth cameras require a more complicated model to correct the 3D measurement volume [50]. In order to reconstruct a scene from the camera pair measurements the system must be calibrated. This includes internal calibration of each camera as well as relative pose calibration between the cameras. A typical approach could be to calibrate the cameras independently and then calibrate only the relative pose between them [51]. In this manner, the calibration of a color camera is a well-studied problem, being the most used approaches based on Zhang's method [52]. However, color discontinuities typically used in calibration approaches are not visible on the depth image. Is then essential to get the best compromise possible in order to find correspondences between the information typically extracted from checkerboard corners present in a color image and the depthmap image of the same scene in which the corners are not visible.

In [50] is proposed a technique that requires the camera to observe a planar pattern shown at a few different orientations. It requires that the pattern is printed and attached to a planar surface. In such approach the checkerboard corners provide suitable constraints for the color images, while the planarity of the points provides constraints on the depth images. The pixels at the borders of the calibration object are ignored and thus depth discontinuities are not needed.

Figure 3.2 shows the different reference frames present in a scene. Points from one reference frame can be transformed to another using a rigid transformation denoted by $\mathcal{T} = \{R, t\}$, where R is a rotation and t a translation. For example, the transformation of a point x_w from world coordinates $\{W\}$ to color camera coordinates $\{C\}$, follows $x_c = R_c x_w + t_c$. Reference $\{V\}$ is anchored to the corner of the calibration plane and is only used for initialization.

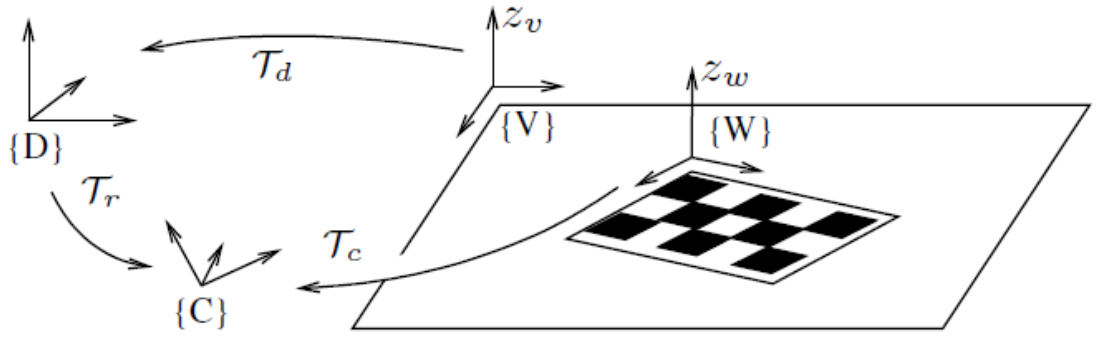

Figure 3.2 - Reference frames and transformations present on a scene. From: [51]

In a concise manner the method proposes an initial estimation for the calibration parameters obtained by independently calibrating each camera. The independent calibrations give an estimation of the transformations $\mathcal{T}_c$ and $\mathcal{T}_d$. The depth intrinsic parameters $\mathcal{L}_d$ and the relative pose $\mathcal{T}_r$ are then refined using a non-linear optimization. Finally, all parameters are refined simultaneously.

3.1.1 Calibration Method

Based upon toolbox released by Daniel Herrera[53] accompanying the work published in [50], previously mentioned Kinect properly calibrated we perform experiments that show an improved accuracy with respect to color/depth alignment seeking to pave the way for fusion and interoperability of the two types of information that Kinect provides. Although the method used is not the only possible solution ([54],[55]) and still presents obstacles to its use, this is one of the most flexible and complete solutions made available.

Worthy of emphasis in this work is the depth correction model, which considers a model based on lens distortion and a new formula to convert disparity values directly measured by Kinect to metric units with direct implications on the result obtained from the alignment. Nonetheless, some repairs may have to be done, mostly related to its lack of automatic segmentation method in the depthmap images used to extract information in order to calibrate the depthcamera, as well as large dependence of the image set provided for training and lack of flexibility of reciprocity possible conversions (RGB→Depth, Depth→RGB).

Figure 3.3 entails some of the effort carried in order to achieve the desired depth color camera calibration while Figure 3.4 comprises some of the conditions necessary to obtain a desirable set of images to provide to the joint depth-color calibration method used. Namely respectable cover the entire viewing area, obtaining data on the overall operating range of the Kinect device and introducing variability on the relative orientation of the reference surface and the camera.

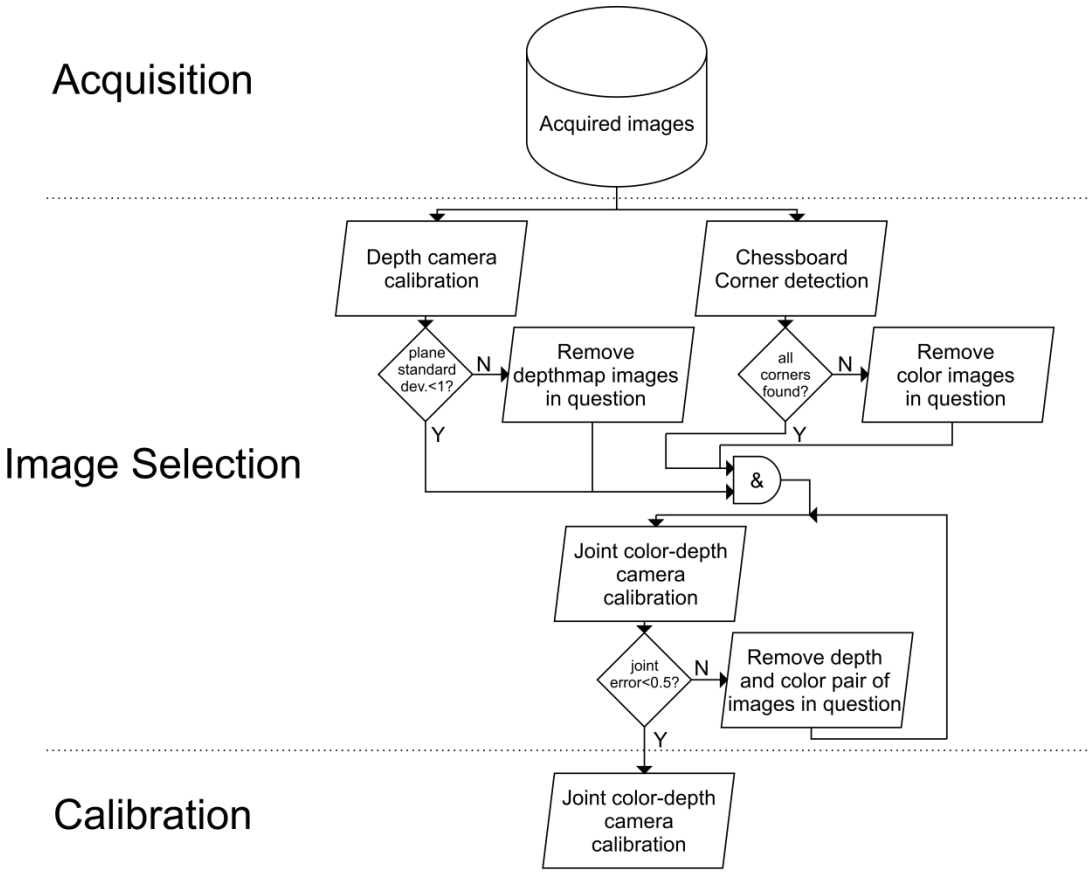

Figure 3.3 - Outline of approach for obtaining the calibration parameters for a Kinect device using the toolbox provided by [53]

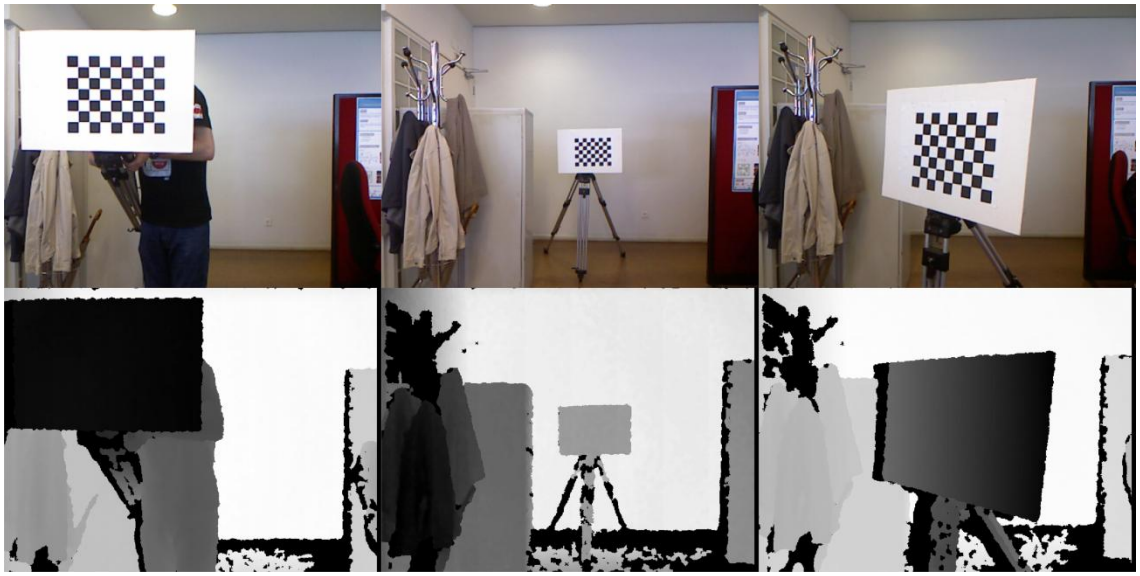

Figure 3.4 - Sample DRGB images for calibration

3.2 Database Construction

There are already several well-established and relevant databases related to animal behavior analysis. A brief review can be found in [49], and in the same paper is introduced a new database worth mention. Nonetheless, as much as possible to ascertain, nothing related to the acquisition with Kinect sensor or something equivalent device was yet published. Thereupon, we started by video recording a singly housed mice, using a Kinect, from a top viewpoint and a standard camera from lateral viewpoint, of both open-field test arena and home-cage housing (Figure 3.5). While Kinect provided both color (8-bit RGB) and depth images (11 bit) in 640x480 resolution at 15 fps using Openni [56], the camera placed laterally provided 640x480 8-bit RGB information at 30 fps.

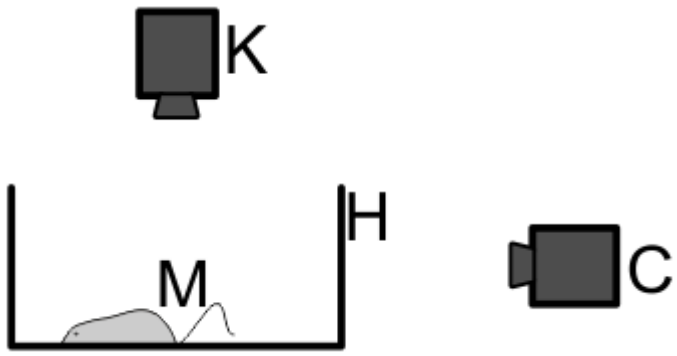

Figure 3.5 - Acquisition set-up consisting of a Canon PowerShot s100 camera (C) placed laterally to the housing apparatus, and a Microsoft Kinect device (K) fixed over the mouse arena (H)

Two mice behaving differently were used for these experiments (Figure 3.6). These videos, corresponding to around one hour of recording (11 videos of about 5 minutes each), were manually annotated labeling every frame of each video sequence with a behavior of interest (Table 3.1): walking, resting, rearing (Figure 3.7), micro movement or other.

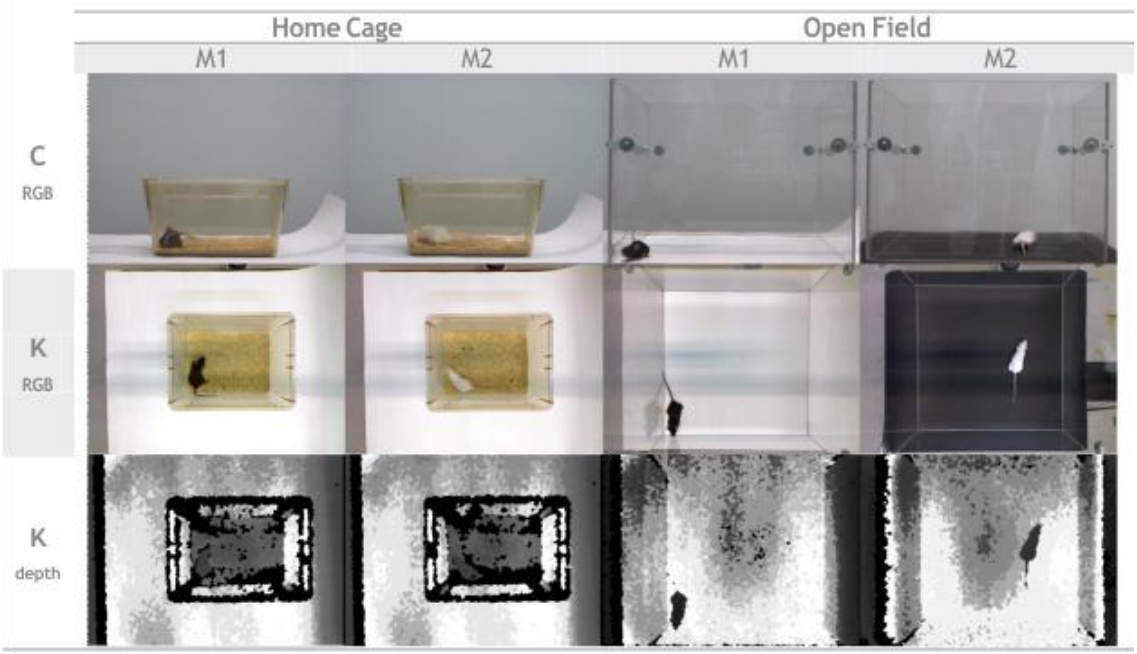

Figure 3.6 - Acquisition examples for two different mouse specimens (M1, M2) and two different housing possibilities (home-cage and open-field Apparatus)

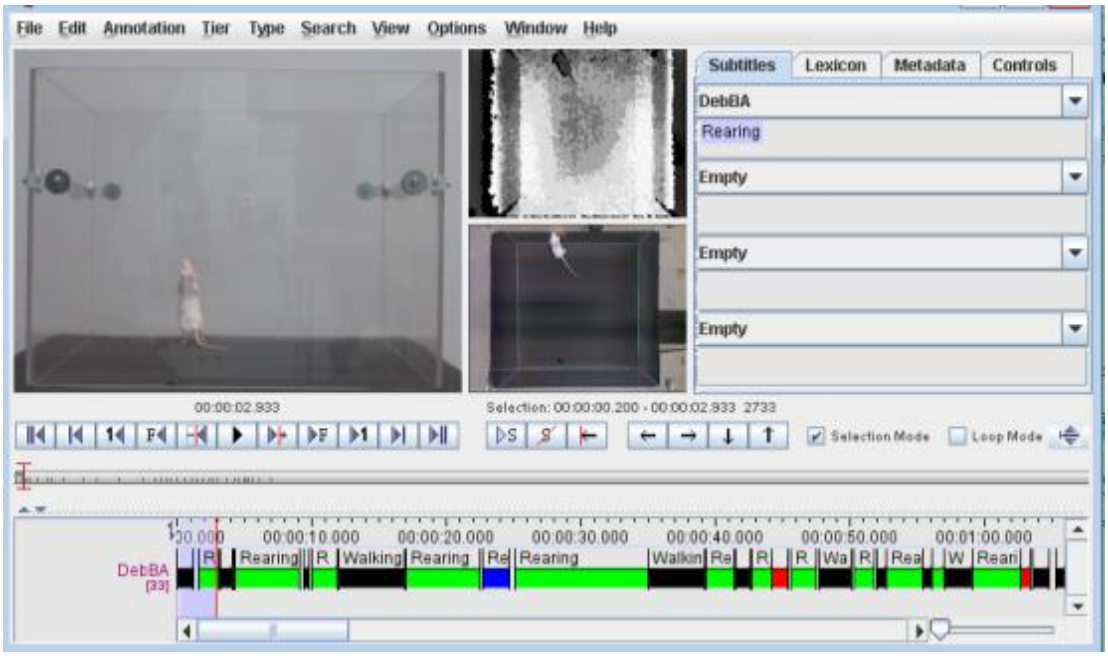

Figure 3.7 – Video annotation sample using European distributed corpora Linguistic Annotator software (ELAN) [57]. Rearing example.

Table 3.1 - Controlled vocabulary of behavior labels

Entries	*Acronym*	*Defined by*
Walking	WAL	ambulation
Resting	RES	inactivity or nearly complete stillness
Micro movement	MIC	sweeping the forelimbs or hind limbs across the face or torso
Rearing	REA	upright posture and forelimbs off the ground
Other	OTH	if no other is found

In order to test several segmentation methods and evaluate depth/color cameras calibration, was performed an additional manual annotation of mice spatial localization in 41 depth map and color images selected from different videos in different conditions of acquisition. Example frames are presented in Figure 3.8.

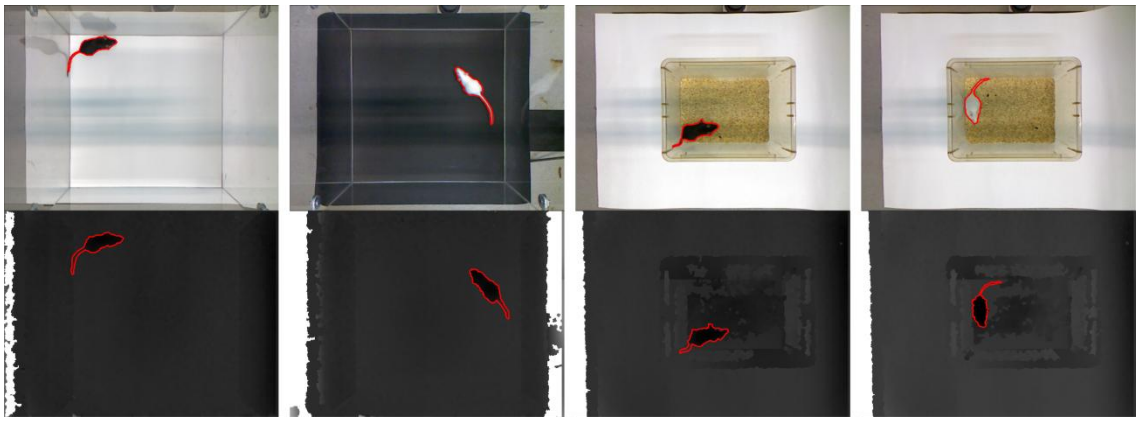

Figure 3.8 – Example of performed manual annotation of mice spatial localization in 41 depthmap and color images selected from different videos in different conditions of acquisition.

3.3 Image Segmentation

Segmentation is set to subdivide an image into its constituent regions or objects of interest. For example, in the context of the present work, interest lies in determining the presence or absence of a mouse on the scene to be analyzed.

Gonzalez [58] advocated that image segmentation algorithms generally are based on one of two basic properties of intensity values:

- discontinuity - based on partitioning an image based on abrupt changes in intensity, such as edges in an image.
- similarity - based on partitioning an image into regions that are similar according to a set of predefined criteria. Thresholding, region growing, region splitting and merging are examples of methods in this category.

As the depthmap from Kinect suffers from inaccuracies at object edges, discontinuity-based may be unadvised and expected to perform poorly. For this reason we focus on the second type of method for segmentation of depthmap images from the Kinect device.

As a gold standard of global thresholding, Otsu's method [59] was used. Other used method was the Gaussian mixture background model (GMBM) [60], which takes into account the pixel's recent history. As a simpler approach it was assumed that the background is static (Static Background Model - SBM). In that way, every new frame is subtracted from the background model previously obtained, and the resulted difference values of each pixel give the information for segmentation. It was also considered local background segmentation approach used in [57] (Local).

3.3.1 Global Threshold

Among all the segmentation techniques, thresholding segmentation method is the most popular algorithm and is widely used in the image segmentation field. The basic idea of automatic thresholding is to automatically select an optimal or several optimal gray-level threshold values for separating objects of interest in an image from the background.

Otsu thresholding technique [59] is one of the global thresholding methods and has been used routinely having proved its effectiveness in a wide range of applications [61]. Concretely, Otsu suggested minimizing the weighted sum of within-class variances of the foreground and background pixels to establish an optimum threshold (T_{opt}). Since minimization of within-class variances is tantamount to the maximization of between-class scatter.

The Otsu method gives satisfactory results when the number of pixels in each class is close to each other. Some limitations of the Otsu method are discussed in [62].

3.3.2 Local Threshold

Simply using a global threshold had proven not yield good results if it occurs variation of acquisition condition within the image itself or if some constraints related to objects sizes are not met. Such remarks led us to take as case of study a recently proposed generic local threshold method implement in particular in Kinect depthmap images [63]. The method proposes that from the depthmap to be analyzed, a density image is defined by transforming the depth information on the depthmap image XY plane to the XZ plane. The value at position (x, z) of the density image denotes the number of points in the depth image at position x (histogram of the column x), taking the value z (by counting along the Y direction).

In this new space, the information on the object of interest is positioned in pixels with low disparity values in the center of the image, while the background is concentrated in the back. A global thresholding method of the original XY image corresponds to defining a horizontal line in the XZ image, discriminating background from foreground. An adaptive thresholding method can be defined as a curve in the XZ image from left to right margins. This results in a threshold that varies from column to column in the original XY image. Since it is necessary for the curve in the XZ image to avoid the parts of the image with high values (high countings), the threshold curve was computed as the shortest path from left margin to the right margin, where the cost of each pixel is its 'intensity' value.

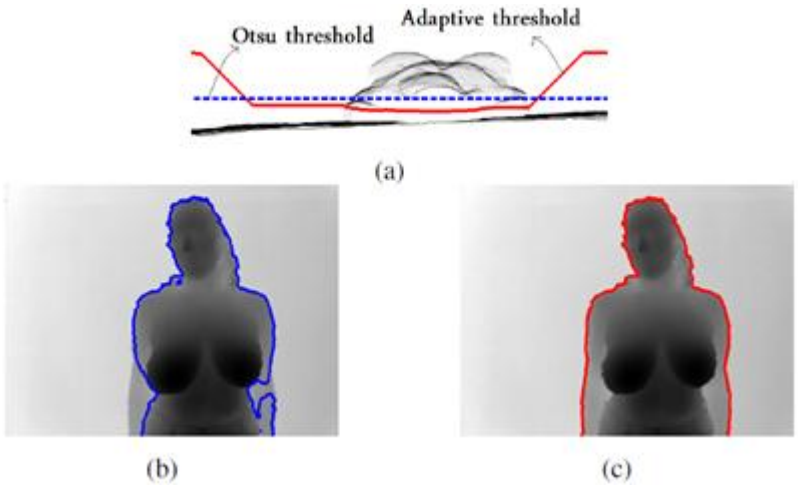

Figure 3.9 - a) XZ plane depth information; b) Otsu's segmentation; c) Local segmentation.

3.3.3 Background Subtraction

It was also considered to model the background taking advantage of the fact that the mouse moves itself through the scene. In a naive approach, given a frame sequence from a fixed camera, the detection of all the foreground objects can be accomplished as the difference between the current frame and an image of the scene's static background:

$$|frame_i - background| > Threshold \quad (2.2)$$

Though simple background subtraction (SBM) has the advantage of computational speed it fails in uncontrolled environments. The most common problems involve temporal background clutter. Mixture of Gaussians is one of most popular approaches although many different methods have been proposed and an exemplary review of techniques published in [64].

The algorithm presented in [60] is representative of an adaptive method which uses a mixture of normal distributions to model a multimodal background image sequence. For each pixel, each normal distribution in its background mixture corresponds to the probability of observing a particular intensity or color in the pixel.

The referred algorithm relies on assumptions that the background is visible more frequently than any foregrounds and that it has modes with relatively narrow variance. These assumptions are consistent with scenes in which the background clutter is generated by more than one surface appearing in the pixel view. Each surface is represented by a normal distribution having a mean equal to the surface intensity or color and a variance due to surface texture, illumination fluctuations or camera noise.

3.4 Feature Extraction

To compute features the assumed approach relied solely in spatial temporal features extrated directly from the segmentation mask for each frame. The following steps describe the method of extraction of the characteristics considered and summarized in Table 3.2 and outlined in Figure 3.11. First step is to fit an ellipse to segmentation mask. The ellipse-based measurements used here were produced by regionprops function from Matlab Image Processing Toolbox based on matching 2nd-order moments [65]. From ellipse extract directly major and minor axis lengths, or mouse length (l) and width (w), as it was considered. Ellipse center of mass ($C(t)$) was used for the calculation of speed $\vec{V}(t) = |C(t) - C(t-1)|$. The velocity is then translated to new mouse coordinate system defined by its minor and major fitting ellipse axis. Finally the angle formed within mouse major axis and the cage floor is computed. For that the mouse point cloud is projected into a plane along z axis and parallel to mouse computed major axis (Figure 3.10).

Table 3.2 – Spatial temporal features to be extracted from depthmap images.

	Symbol	*Meaning*
1	θ	Angle of elevation of the mouse
2	w	Mouse width
3	l	Mouse length
4	V_y	Lateral velocity (velocity component within mouse fitting ellipse minor axis)
5	V_x	Principal velocity (velocity component within mouse fitting ellipse major axis)

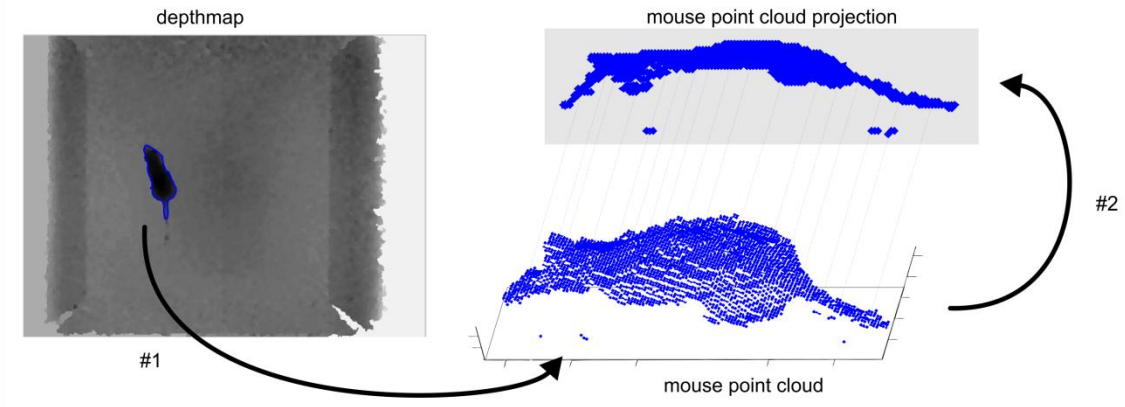

Figure 3.10 – Mouse lateral projection image.

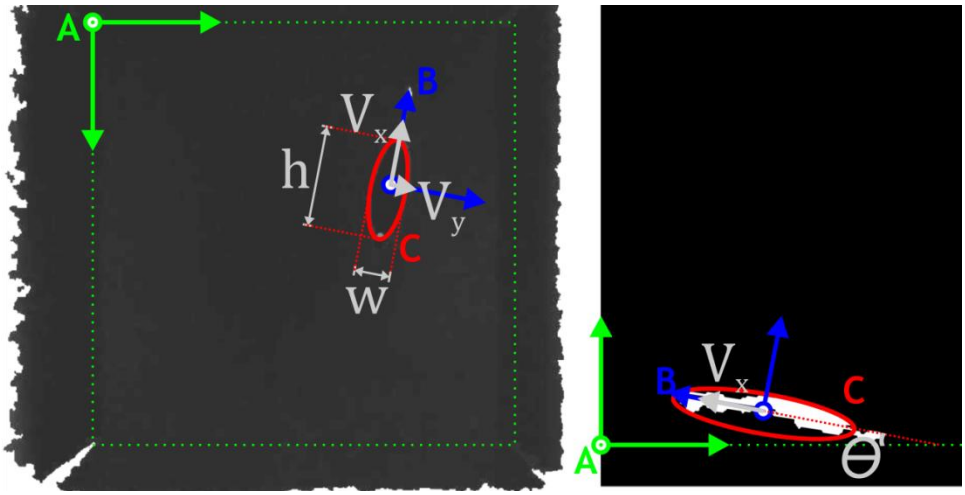

Figure 3.11 – Feature extraction in both depthmap image and mouse point cloud lateral projection.

3.4.1 Feature Verification

Some assumptions were established for a mouse model at this point in order to ensure that possible faults regarding segmentation or the acquisition conditions in specific frames were not critical for the feature extraction process. A possible cause of error is shown in Figure 3.12. A small segment is shown in which the position of the mouse is almost perpendicular to the camera and hence parts of the body do not allow recovery of the depth value by Kinect, (black areas in the image).

Assumptions were established over mouse velocity and length/width ratio. Adicional annotation work was performed for a representative video, having been manually annotated information on the location of the tail head and center of mass. The annotation information provided the reference values for establishment maximum velocity, and minimum mouse length/width ratio. For each new frame the computed features are compared with corresponding references and the frame labeled as uncertain.

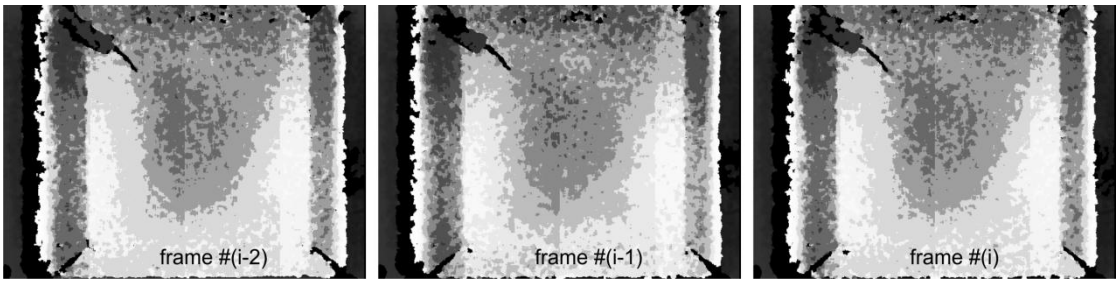

Figure 3.12 - Example of incomplete depth map situation for a particular mouse pose.

3.4.2 Tracking

Closely related with the way the features are extracted it is proposed an approach to efficiently track the mouse extremities trough a video. The proposition assumes that for a particular video sequence, a mouse does not reverse its direction between consecutive frames. In that way, once the mouse is successfully segmented and its major axis known, one can make use of the angles measured between the mouse major axe and the image horizontal axe, $\theta_i \epsilon]-180^o, +180^o]$ (Figure 3.13), to establish a correspondence between the extremities, in the following way: in the first frame, establish the smallest possible angle in the selected reference, as the orientation of the mouse; for subsequent frames, consider both possible angles to measure, choose for mouse orientation the one whose absolute difference with the previous angle is smaller.

This approach has the benefit of allowing the establishment of correspondence between extremities of mice, even when in each frame there are not sufficient discriminative features.

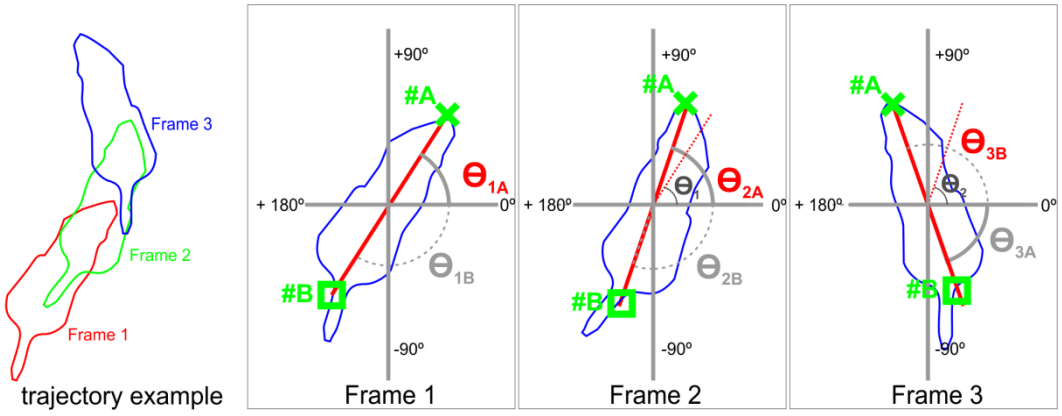

Figure 3.13 – Illustrative example of mouse tracking approach.

However, the simple rule described above, does not allow to distinguish which extremity is mouse head or tail. For that more prior knowledge was used. The formulated premise establishes that although mouse may present negative principal velocity, V_x, the mouse does not walk backwards, in a stable and continuous way in time. In this way, the orientation that has been assumed by the earlier method is marked as *forward* or *backward* depending on the verification of a principal velocity component, consecutively positive or negative on a typical time window, for the movement of the mouse.

3.5 Classification Algorithms

Given a set of m examples $z_i = \{(x_i, y_i)\}_{i=1}^{m}, x \in \mathcal{X}, y \in \mathcal{Y}$ drawn from a joint distribution $\mathcal{P}$ on $\mathcal{X} \times \mathcal{Y}$, the ultimate goal of the learning algorithm is to produce a classifier function $f: \mathcal{X} \to \mathcal{Y}$ such that the expected error of f, given by the expression $E_{(x,y)\sim\mathcal{P}}(f(x) \neq y)$, is minimized. The examples $x_i \in \Re^n$ are vectors of n features, which may equivalently be referred to as measurements or variables. These features can be raw, low-level measurements, such as the intensities of individual pixels in an image, or higher-level representations such as edge-filter outputs or cross-correlation scores. Some feature representations elicit better classification performance than others, but otherwise there is no restriction stating what makes an appropriate feature value [32].

The most common approach to classification is therefore to use a combination of spatio-temporal features with a classifier such as Support Vector Machine (SVM), Naive Bayes, k-Nearest Neighbour (kNN) or Neural Networks [66]. Although a large number of techniques have been developed we will focus on one of the most popular, yet simple, algorithm for classifying data of various types into prescribed categories: decision trees. It offered the best compromise of computational time considering preliminary classification study carried out in this dissertation.

Decision trees are commonly used in classification problems with categorical data, and in [67] can be found an overview of work in decision trees.

Concisely, decision trees classify instances by sorting them based on feature values. Each node in a decision tree represents a feature in an instance to be classified, and each branch represents a value that the node can assume. Instances are classified starting at the root node and sorted based on their feature values.

Decision trees construct a tree of questions to be asked of a given example in order to determine the class membership by way of class labels associated with leaf nodes of the decision tree (Figure 3.14). This approach is simple and has the advantage that it produces decision rules that can be interpreted by a human as well as a machine [68].

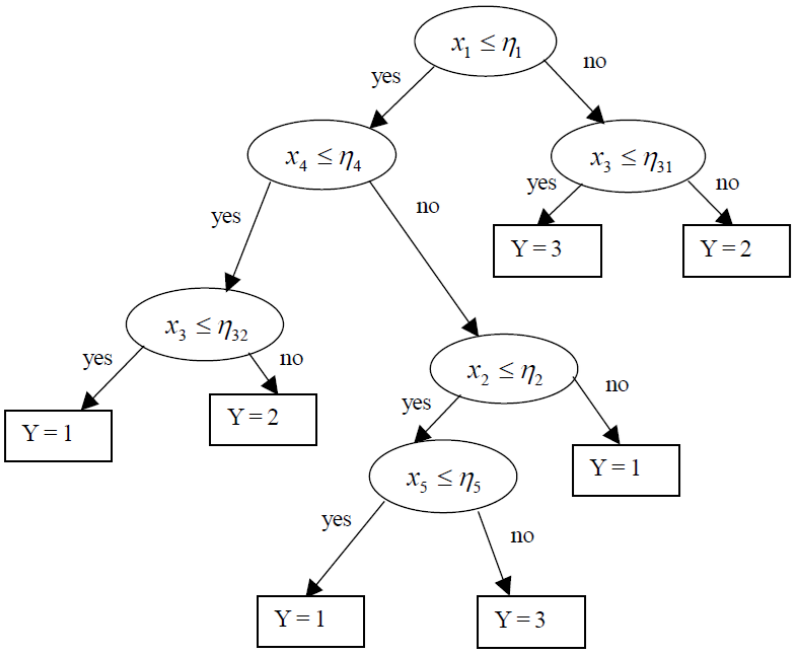

Figure 3.14 - A classification tree for a five dimensional feature space and three classes. The x_i are the feature values, the η_i are the thresholds, and Y is the class label. Retrieved from [69].

Chapter 4

Results and Discussion

On this section, some results concerning the mouse segmentation methods for depthmaps images, color and depth cameras calibration, mouse tracking and behavior recognition are presented. These results provide the necessary data to evaluate and to discuss the possibilities provided by simultaneous color and depth information acquisition, for mouse behavior recognition purposes. This also enables the identification of weaknesses that need improvement in order to achieve a functional behavioral analysis system.

4.1 Calibration

In order to evaluate depth/color information alignment, as exemplified in Figure 4.1, the depthmap image ground truth (DGT) is compared with each one of these three informations:

- Ground truth of mouse location manually segmented in the RGB image (DGT);
- Ground truth resulting from transporting the DGT to the depthmap space using the calibration parameters resulting of the calibration procedure already explained, performed for one Kinect device provided by INESTEC (DGT1);
- Ground truth resulting from transporting the DGT to the depthmap space using the calibration parameters performed for the same Kinect device used (DGT2);

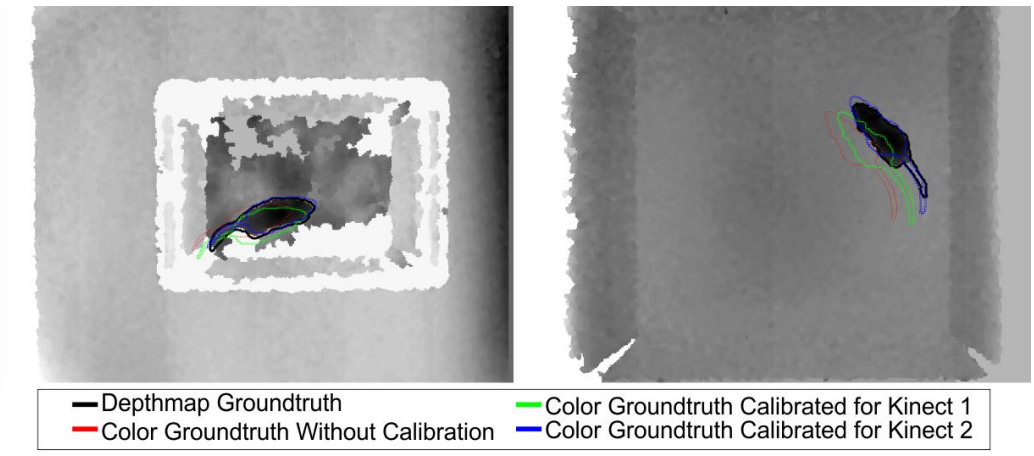

Figure 4.1 - Examples of the evaluation of the alignment performed for 2 images in a total of 41 cases.

The quantification of the quality of the alignment is shown in Table 4.1 both Hausdorff and mean distance between ground-truth boundaries acquired for both depth and color images. In detail, the Hausdorff distance [70] is defined as

$$H(A,B) = \max(h(A,B), h(B,A)) \tag{4.1}$$

$$h(A,B) = \max_{a \in A} \min_{b \in B} \|a - b\| \tag{4.2}$$

while the mean distance is computed such as

$$\mu(A,B) = \text{mean}(\text{m}(A,B), \text{m}(B,A)) \tag{4.1}$$

$$m(A,B) = \text{mean}_{a \in A} \min_{b \in B} \|a - b\| \tag{4.2}$$

given two finite point sets $A = \{a_1, \dots, a_p\}$ and $B = \{b_1, \dots, b_q\}$, and $\|\cdot\|$ being the L_2 Euclidean norm on the points of *A* and *B*.

Table 4.1 - Depth/color maps alignment evaluation. Hausdorff and mean distances between ground-truth masks for a set of 41 depth/color image pairs manually annotated

Without calibration		*Calibration parameters 1*		*Calibration parameters 2*	
H(RGB,D)	μ (RGB,D)	H(RGB,D)	μ(RGB,D)	H(RGB,D)	μ(RGB,D)
27.7 ± 9.4	10.7± 6.3	23.3 ± 7.6	9.7 ± 4.3	12.8 ± 8.1	5.0 ± 3.9

As the reader may attempt, the results of the Table 4.1 confirms what the Figure 4.1 has left foresee. On one hand, we corroborate the need for calibration of Kinect in order to make use of information correspondence, or even interchangeability between color spaces and depth. On the other hand, we illustrate the existence of differences between devices, even if only on the level of required quality control of a device within the range of prices of Kinect, but which results in the calibration parameters change (which includes lens correction model, and parameters intrinsic to each of the cameras) and consequently a poor alignment result when transforming the images with resulting calibration parameters of a device different from the images under analysis. An example of both color and depth information properly aligned is presented in Figure 4.2 and Figure 4.3.

Figure 4.2 - Kinect point cloud and color information merged example

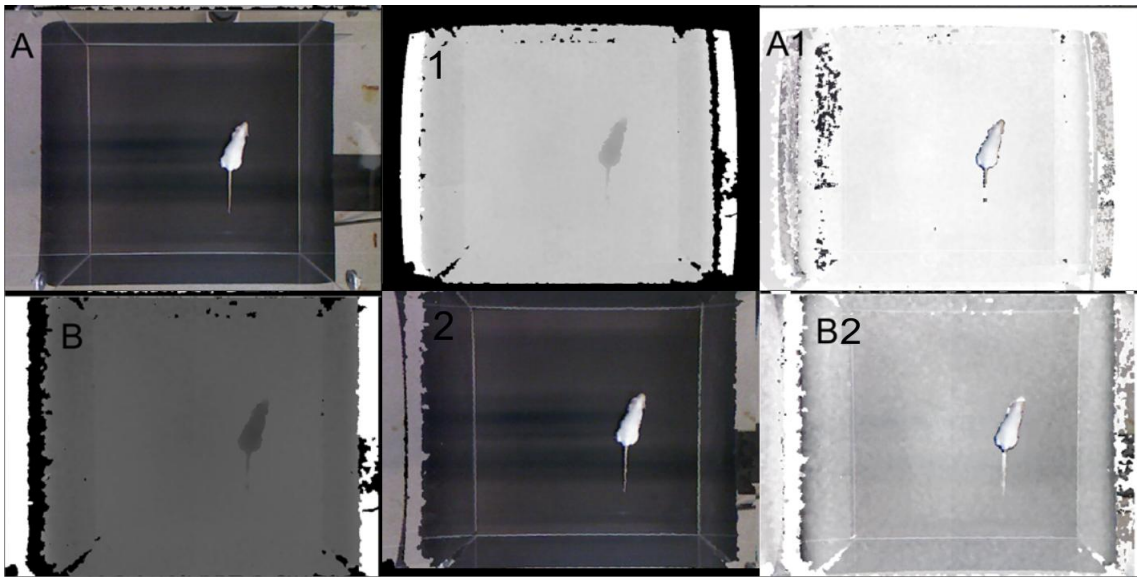

Figure 4.3 - Example of depth color information matching. A: RGB image from Kinect stream; B: Depthmap image from Kinect stream; 1:Depthmap image transported to RGB space; 2: RGB image transported to depthmap space; A1 and B2: Depthmap/RGB alignment example.

4.2 Image Segmentation

The effort realized on depthmap image segmentation is discussed here. Its goal was to robustly be able to extract masks surrounding the mice against the involving arena. Before turning to the results it is important to do here a disclaimer on the measures employed. While the true positives (TP) give the number of correctly detected foreground pixels, the true negatives (TN) give the number of correctly identified background pixels. In contrast the false negatives (FN) are pixels that are falsely marked as background, whereas false positives (FP) are falsely detected as foreground. Hereupon, the measures computed were the true positive rate (TPR), given by

$$TPR = \frac{TP}{TP + FN} \tag{4.1}$$

and the false positive rate (FPR), given by

$$FPR = \frac{FP}{FP + TN} \tag{4.2}$$

Table 4.2 shows that the best results were verified for SBM method due to the existence of a general stability of the background in the images analyzed.

Table 4.2 - Results for depth based mouse segmentation for the 41 depthmap images manually segment (in % pixels).

	Segmentation methods			
	GMBM	***Otsu***	***Local***	***SBM***
True positive rate	0.30	1.00	0.56	0.68
False positive rate	0.00	0.95	0.02	0.00

According to the published work [57] the shortest path that allows establishing the threshold value to each column is set to be determined depending on histogram frequency only. Also, objects of interest corresponded to a more obvious situation of two large and distinct groups. Particularly, Figure 4.4 which understands a typical open-field apparatus situation meets with the situation and it behaves well. The problem arises in situations such as presented in Figure 4.5, where it can be attempt a situation of multiple backgrounds. The Local method showed issues dealing with complex backgrounds and uniformity of cost criteria for different situations as open field and home-cage arenas dictated their poor results.

Since in our samples, the mice area was much smaller than the background, the presence of different objects at different depths (substrate materials covering the bottom of the cage) caused wrong classifications by Otsu method. Occasional occurrences of long periods of immobility lead to the failure segmentations by GMBM method.

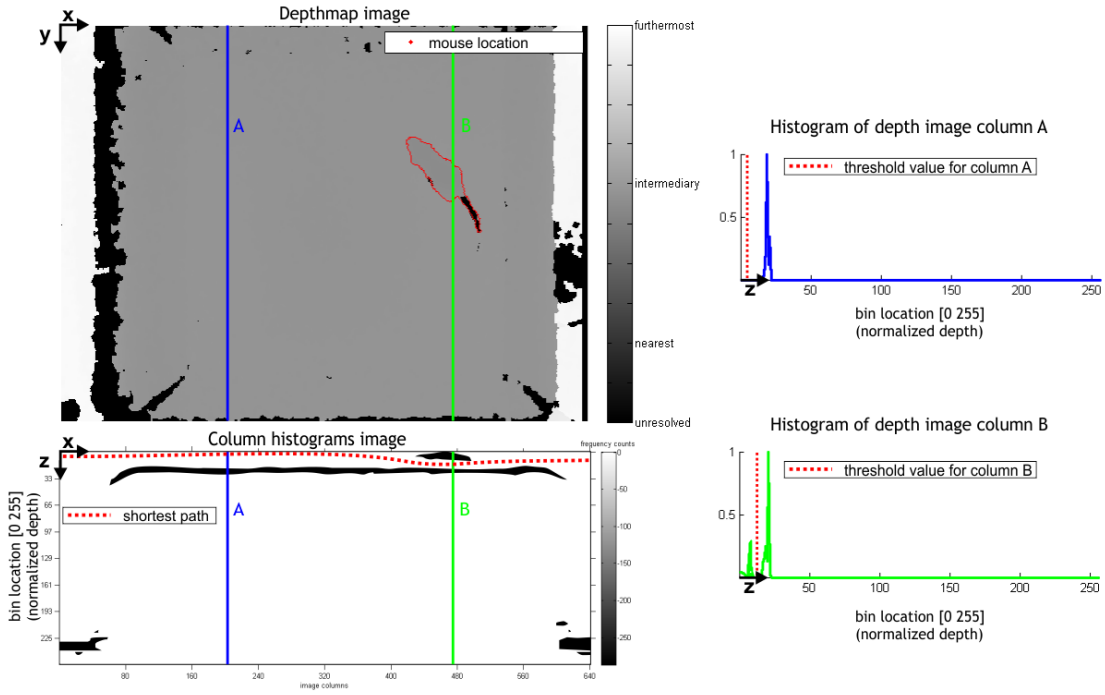

Figure 4.4 - Local threshold method based on shortest path in the column histograms image example for the open-field apparatus.

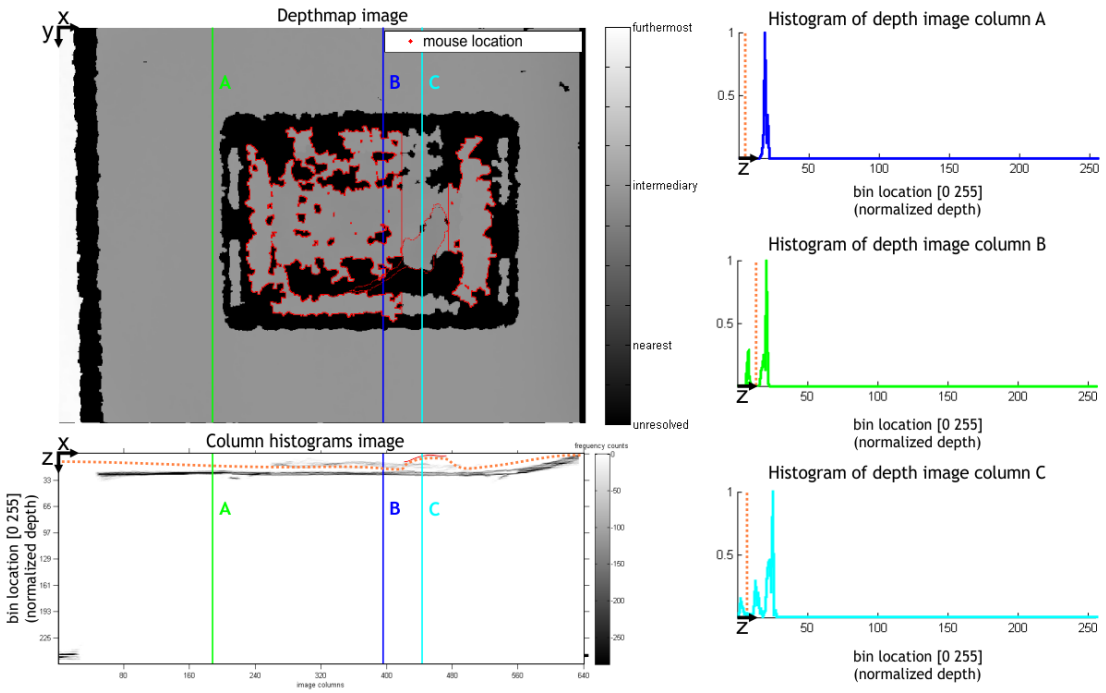

Figure 4.5 - Local threshold method based on shortest path in the column histograms image example for the home-cage apparatus.

4.3 Tracking and Feature Extraction

Already verified in different works (as in [71] or more recently in [72]) Kinect present some challenges regarding returning depth information of small irregular and that pose almost perpendicularly in respect to the camera. As explored in Appendix A, Kinect's depth sensor relies on the comparison of a known pattern against each new frame retrieved from its infrared camera that is supposed to contain a specular pattern projected by Kinect itself and deformed by real world scene. The transparency, specularity, and reflection of projected dot pattern are therefore situation to avoid. Moreover, the depth sensed at the border of small objects is very noisy, because the depth resolution sensed is in order of centimeters [73].

One may attempt in Figure 4.6, Figure 4.7 as well as the Figure 3.12 already mentioned, as same examples of how the application of Kinect device to the problem the mice behavior recognition poses.

Although naïve, the segmentation method SBM combined with the feature verification metrics was able to keep tracking of the mice in the tested videos (around one hour of recording as the reader may recall from the 3.5 Section). It presented a hit rate of 100% (Number of times the center of mass of segmentation mask was successfully detected inside a manually annotated bounding box referred to the number of frames). Nonetheless the segmentation was found loosely compromised and incomplete, especially in the home-cage scenario. In average in the videos from our database a frame from home-sage apparatus (Figure 4.6 (left)) presented 19.38% of its area with unresolved pixels. Some work is already being developed in order to overcome some of the situations described. In [74] point us to the use of multiple Kinect devices. Another possible path to be followed could be further work in the field of reconstruction. And may accordingly the work published in [75] serve as starting point.

Nonetheless, in the scope of this thesis, looking to evolve to exploration of the field of learning and recognition of behavior itself, we chose to focus attention on the open-field situation. Although presenting in our videos an average of 7.93% of unresolved areas and situations such as in Figure 4.7 cannot be disregarded, it corresponds to a more manageable situation, yet desirable and significant for the purpose of advancing animal behavior recognition systems.

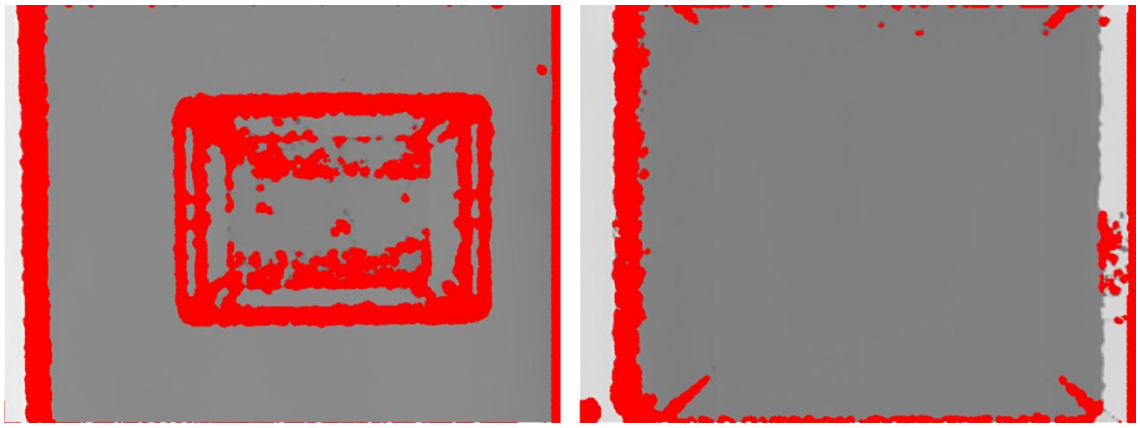

Figure 4.6 - Sample depthmap images of home-cage (left) and open-field (right) apparatus with unresolved areas marked red (pixels for which the Kinect does not return any depth value).

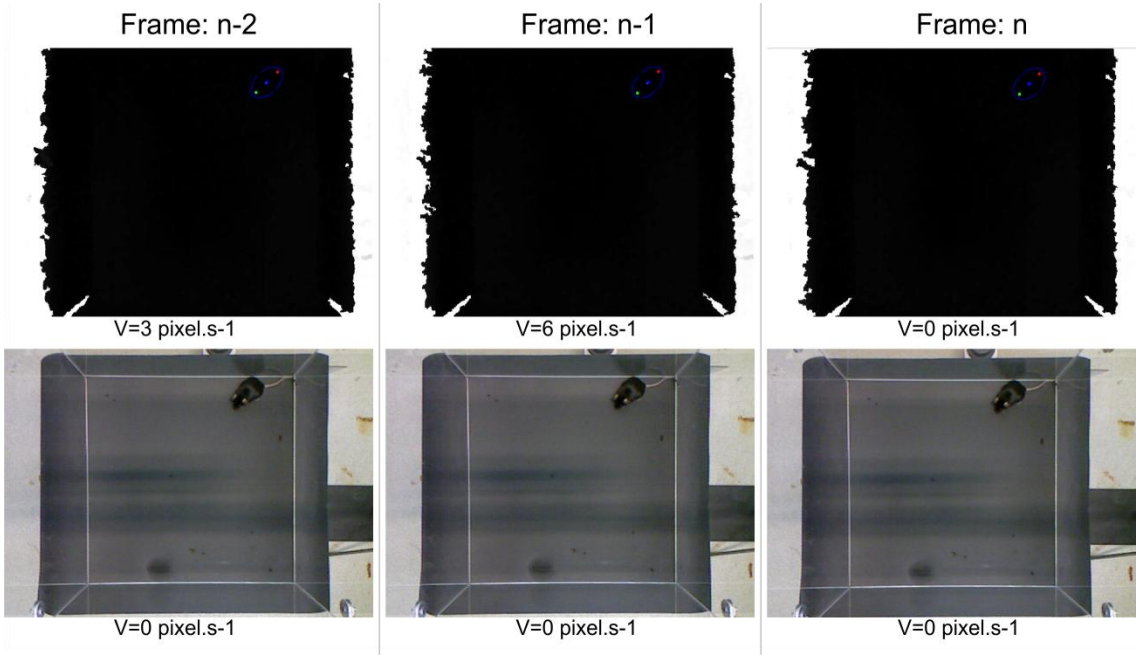

Figure 4.7 - Example of noise introduced by Kinect due to its low resolution and specularity problems.

4.4 Classification

In this section is present an evaluation of the relative importance of different spatial temporal features. It is also explored the introduction of time windows of interest that would establish temporal context. Now focusing attention just in the situation of open-field apparatus and considering the features (summarized in Table 3.2) extracted from 4 videos (~6 min each) relying on the segmentation approach that scored best in the comparative study on segmentation technics for depthmap images.

Going back to fulfill our first challenge, we studied the impact of each feature by different training classifiers considering all possible combinations of all features for each case of possible number of features. Performance was estimated using a cross validation procedure, whereby all but two of the videos were used to train the system, and performance was evaluated on the remaining videos. The result of the best combination for a number of different features is presented in Table 4.3.

Table 4.3 - Best feature combinations - misclassification error using decision tree (comparison against manual annotation using cross validation).

	Number of features				
	1	2	3	4	5
misclassification	0.448±0.10	0.358±0.14	0.352±0.09	0.350±0.10	0.346±0.11
features	$\{V_x\}$	$\{V_x, \theta\}$	$\{V_x, \theta, l\}$	$\{V_x, \theta, l, V_y\}$	$\{V_x, \theta, l, V_y, w\}$

Figure 4.8 ilustrates one of the pruned decision trees trained during the evaluation of features quality. All seem to contribute for the improvement of overall recognition of the the set of five behaviours against manual annotations althougth principal component velocity and elevation angle contribuition stand out from the other features. Nonetheless, as the Figure 4.9 can clarify to some extent, these instantaneous spatiotemporal features themselves seem not to close the solution to find the function that discriminate the desired behaviors.

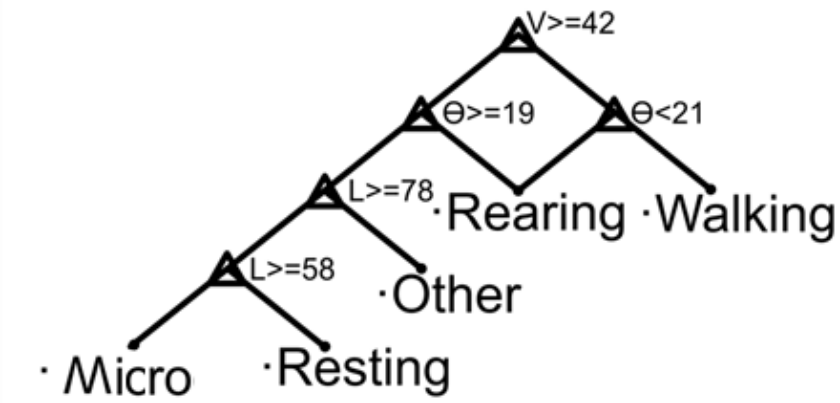

Figure 4.8 - Example classification based on a decision tree with three feature (mouse length (L, pixel), angle of elevation (θ, °), principal velocity (V, pixel.s^{-1})).

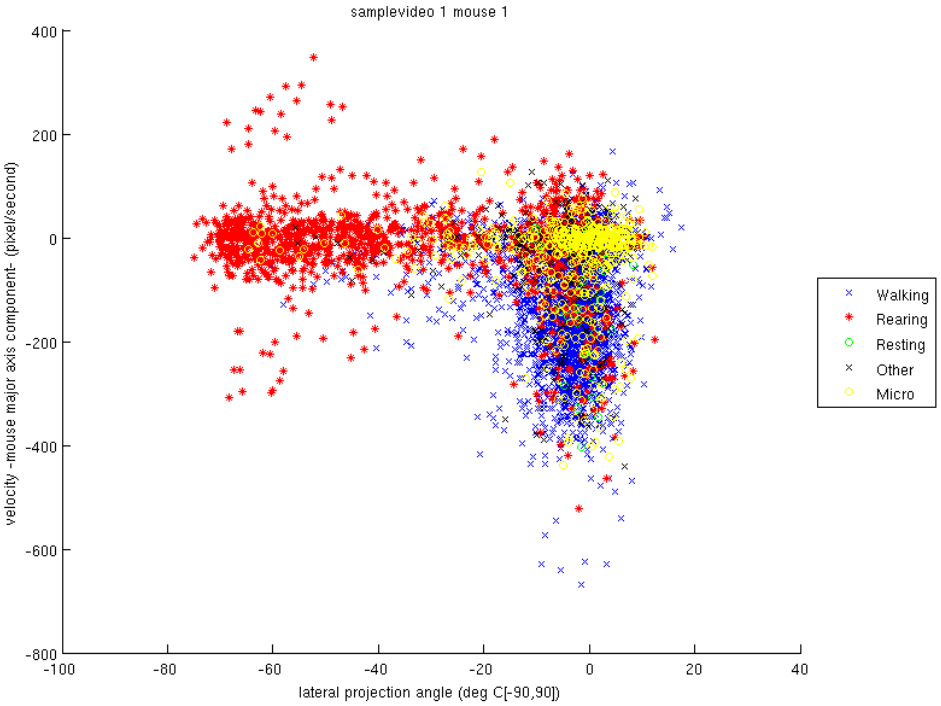

Figure 4.9 - Principal velocity and elevation angle distribution.

Keeping the decision trees tool, it was pursued to establish a temporal context to the features. Instead of a set of simple measures from the instant frame, or in a window as narrow as the previous frame we studied the addiction of features that comprise previous values promoting a notion of weak trajectory and movement history. It was used the same evaluation as described above and conducted the experiment for each feature individually. Window size for each feature was selected as a resulting misclassification error minimum, as marked in Figure 4.10.

The evaluation of the simultaneous use of different time windows for each of the features is shown in Table 4.5 immediately after the Table 4.4 with results for previous considered situation of 5 features. Both tables present the results for the best combination of two videos for training and testing for other videos. Although the global results improve is not achieved the pursue of improving the significantly poor results obtained for classification of the behavior of micro-movement.

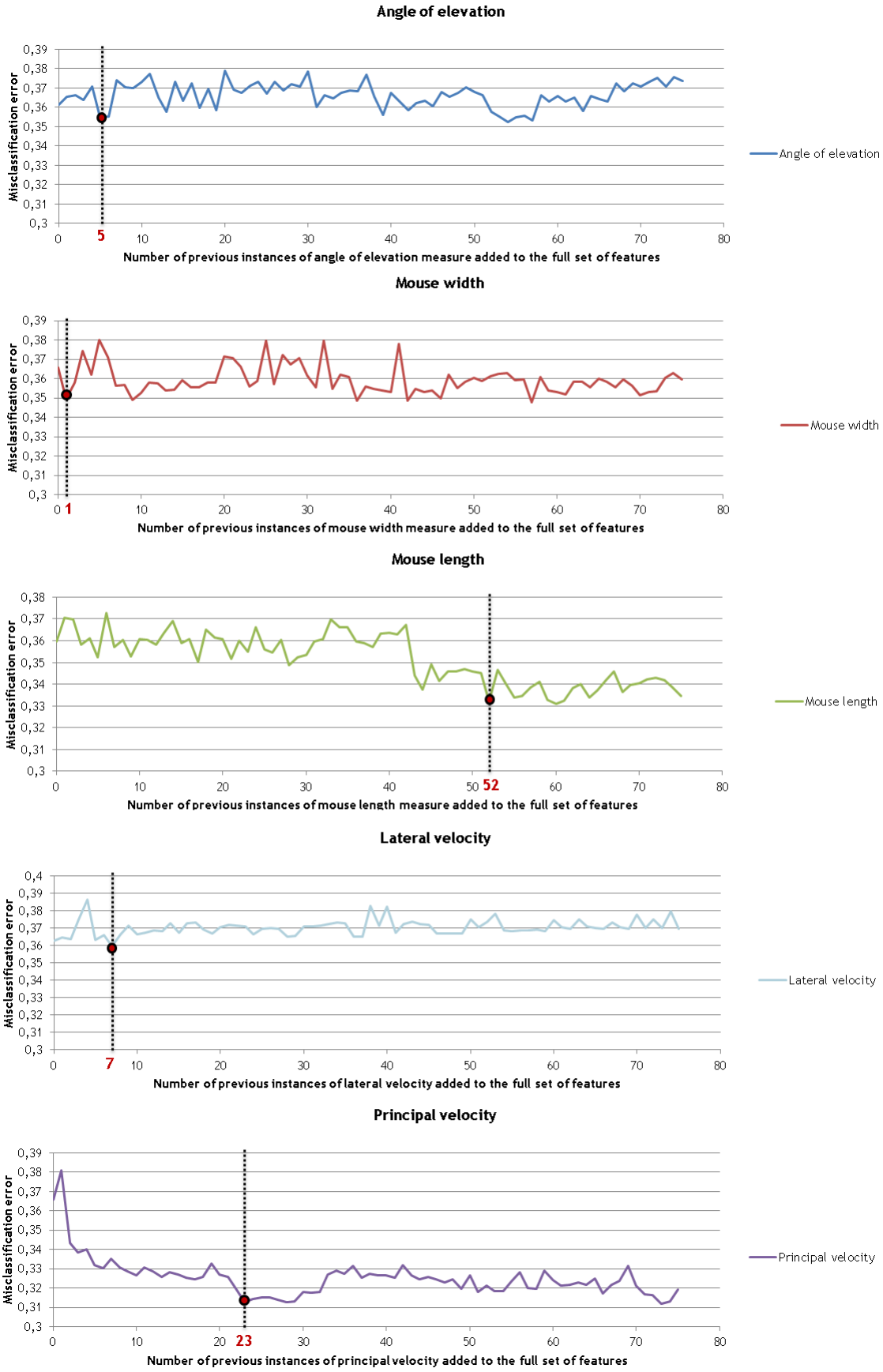

Figure 4.10 - Study for establishing a time window of interest for each of the features.

Table 4.4 - Confusion matrix for best decision tree with five features against manual annotation in two videos of about 5min each.

		Predicted class				
		Walking	Rearing	Resting	Other	Micro
Actual Class	Walking	0.773	0.044	0.070	0.074	0.039
	Rearing	0.177	0.632	0.051	0.085	0.054
	Resting	0.068	0.001	0.824	0.082	0.025
	Other	0.260	0.049	0.376	0.194	0.121
	Micro	0.309	0.081	0.179	0.086	0.345

Table 4.5 - Confusion matrix for best decision tree with 93 features evaluated against manual annotation in two videos of about 5min each (5 features and its values in the past instances using the time windows illustrated in Figure 4.10).

		Predicted class				
		Walking	Rearing	Resting	Other	Micro
Actual Class	Walking	0.809	0.030	0.035	0.086	0.041
	Rearing	0.159	0.685	0.009	0.097	0.050
	Resting	0.050	0.000	0.794	0.083	0.073
	Other	0.252	0.048	0.316	0.210	0.174
	Micro	0.291	0.074	0.097	0.151	0.387

4.5 General Discussion of Results

Throughout the results presented we established a potential method for recognition of behavior for a singly housed mouse in an open-field apparatus capable of identifying walking, resting, rearing and micro-movement occurrences. It relies on the information acquired by a Kinect device, previously calibrated, from which are extracted spatial temporal features which are passed to a previously trained classifier being possible to assign each frame a behavior within the set considered while training the classifier.

Although, with the data presented in this work, no fare comparison with existing systems can be established some remarks can be achieved. The best performance results with a cross validation on 4 videos (corresponding to about 25 min) for the recognition of 4 behaviors (excluding *other* from the total 5) using only spatial temporal features was 66.9%. That can be considered worth notice especially bearing in mind the typical agreement between different annotators (Figure 2.5 (b) and Figure 2.7 (a)) around 70%.

On the other hand the results for micro movement actions in particular are somehow disappointing. However two considerations on the subject can be made. First is the least represented behavior (more in Appednix B) and tree decision may not be the best solution for dealing with such situation. Second of all it may be noted that is also the behavior that performs worth in the comparison between different annotators. It can actually be distinguished from the others behavior studied here for belong to a kind of high-speed model

behavior as is suggested in [76], where a solution was proposed using a 240-fps image analysis system. Although other approaches had been trying to tackle the challenge differently (new sets of descriptors and advanced context analysis as example) it can also be the case that acquiring full resolution provided by Kinect at 15 fps does not guarantee the Nyquist-Shanon sampling theorem for the purpose of micro movement recognition.

Also worth mention the solution for mice tracking based on the naïve approach of the static background model. Although in a constrained way (by means of ensuring the existence of a known static background) it can be applied to a wider set of situations than systems such as [21] that depend not only on a static background scenario but a particular set of animals and background colors.

Chapter 5

Conclusions and Future Work

Though the use of automated approaches has been documented for mice behavior recognition, such systems are not widely used because they present limitations and may be cost prohibitive.

The simple framework presented proved to be able to extract features from the Kinect input device, and use those for the automated recognition of the animal's behavior by a statistical supervised classifier. The main focus was centered on the adaptation of a device yet relatively new, to a problem for which many people have contributed with solutions, not meaning that there are no issues that still need to be resolved. The results fall short at any given time by the small database established and by some simplicity of the methods explored with respect to validate unequivocally a novel method for the recognition of behavior. Nonetheless it can be unhesitatingly affirmed that several noteworthy preliminary results have been achieved successfully.

Segmentation could be further improved. However, some interesting results are presented. Considering the restrictions in the tested data, the naïve approach of a static background model proved itself useful. It is important to reinforce the effort that must be continued in order to expand the segmentation step to a less constrained environment. Stressing on this fact a major focus of interest should be given to Kinect due to much lower dependence of lighting color restrictions.

Also worth noting the fact that although presented behavior recognition results are not enough for the proof of the unequivocal benefit of Kinect yet, we can point to a certain positive outlook. In [49] from which was extracted Figure 2.6 was demonstrated the benefit of using the information acquired simultaneously from top and side view through the use of two video cameras. Though in a different field of application and so results in itself should not be directly compared with the method presented here. It is important to emphasize the fact that Kinect requires clear path to the scene. It is still worth highlighting, as shown, to be possible to have three-dimensional information in a less costly way and wait for further developments to definitively endorse it.

It may also be noted that depth map images from Kinect are typically noisy and incomplete, mostly due to occlusion, relative surface angle and material, making us to

consider Kinect RGB images an essential complement to depth maps, as well as more work regarding improving depthmap quality itself.

5.1 Future Work

Further work must be done in order to attest the importance of using additional information provided by depth maps, for the behavior recognition task. Namely, inclusion of velocity and position features computed from depth map segmentation mask in a more matured behavior recognition framework and evaluation of its relevance to the method performance.

Regarding recognition algorithm for animal behavior analysis, some work could be done in particular through use of more evolved learning tools, such as Support Vector Machines (SVM) technique. The SVM [77] is an algorithm that locates a decision boundary between the two classes of examples in multidimensional space, such that the margin is maximized. The examples closest to the boundary are called support vectors and their distance from the boundary is the margin. Depending on the type of kernel used, the decision boundary can take on many different shapes, from an *n*-dimensional hyper plane in the linear case, to a complex, bumpy surface in the case of a Gaussian kernel.

Particularly, one of original Support Vector Machine classifier developed in the 90's, for sequence tagging extensions worth mention is a hidden Markov Support Vector Machine that uses a temporal model [78]. Such extension, models dependencies between each label y_i and each input $x = \{x_1, \dots, x_l\} \in \Re^n$, as well as dependencies between y_i and y_{i-1}, and has been already used in training classifiers to recognize behaviors of mice [21].

Also not to forget the existence of other spatial temporal point descriptors [8] and advanced context analyzers [49], but also general improvement and generalization in order to robustly tracking the mice in a less constrained manner.

References

[1] International Human Genome Sequencing Consortium, “Initial sequencing and analysis of the human genome,” *Nature*, vol. 409, no. 6822, pp. 860–921, Feb. 2001.

[2] E. S. Lander, “Initial impact of the sequencing of the human genome,” *Nature*, vol. 470, no. 7333, pp. 187–197, 2011.

[3] J. F. Cryan and A. Holmes, “The ascent of mouse: advances in modelling human depression and anxiety,” *Nat Rev Drug Discov*, vol. 4, no. 9, pp. 775–790, 2005.

[4] H. J. Hedrich and G. R. Bullock, *The laboratory mouse*, 1st ed. Academic Press, 2004.

[5] B. M. Spruijt and L. DeVisser, “Advanced behavioural screening: automated home cage ethology,” *Drug Discovery Today: Technologies*, vol. 3, no. 2, pp. 231–237, 2006.

[6] O. H. Maclin and M. K. Maclin, “Coding observational data: A software solution,” *Behavior Research Methods*, vol. 37, no. 2, pp. 224–231, May 2005.

[7] S. Y. Elhabian, K. M. El-Sayed, and S. H. Ahmed, “Moving object detection in spatial domain using background removal techniques-state-of-art,” *Recent patents on computer science*, vol. 1, no. 1, pp. 32–54, 2008.

[8] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie, “Behavior recognition via sparse spatio-temporal features,” in *2nd Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance, 2005*, 2005, pp. 65– 72.

[9] T. B. Moeslund, A. Hilton, and V. Krüger, “A survey of advances in vision-based human motion capture and analysis,” *Computer Vision and Image Understanding*, vol. 104, no. 2–3, pp. 90–126, Dec. 2006.

[10] H. Jhuang, T. Serre, L. Wolf, and T. Poggio, “A Biologically Inspired System for Action Recognition,” in *Computer Vision, IEEE International Conference on*, Los Alamitos, CA, USA, 2007, vol. 0, pp. 1–8.

[11] B. Vemuri, A. Mitiche, and J. Aggarwal, “Curvature-based representation of objects from range data,” *Image and Vision Computing*, vol. 4, no. 2, pp. 107–114, May 1986.

[12] F. Arman and J. K. Aggarwal, “Model-based object recognition in dense-range image - a review,” *CSUR*, vol. 25, no. 1, pp. 5–43, Mar. 1993.

[13] K. W. Bowyer, K. Chang, and P. Flynn, “A survey of approaches and challenges in 3D and multi-modal 3D+2D face recognition,” *Computer Vision and Image Understanding*, vol. 101, no. 1, pp. 1–15, 2006.

[14] E. Stoykova, A. Alatan, P. Benzie, and et al, “3-D time-varying scene capture technologies,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 11, pp. 1568–1586, 2007.

[15] Microsoft Corporation, “Kinect - Xbox.com.” [Online]. Available: http://www.xbox.com/en-US/kinect. [Accessed: 17-Jan-2012].

[16] ASUSTeK Computer Inc., “ASUS Xtion PRO,” *ASUS Multimedia Motion Sensor*. [Online]. Available: http://www.asus.com/Multimedia/Motion_Sensor/Xtion_PRO/. [Accessed: 17-Jan-2012].

[17] SOFTKINETIC, “DepthSense 311 camera.” [Online]. Available: http://www.softkinetic.com/Solutions/DepthSensecameras.aspx. [Accessed: 17-Jan-2012].

[18] B. Liefeng, R. Xiaofeng, and F. Dieter, “Depth kernel descriptors for object recognition,” in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2011, pp. 821–826.

[19] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, “Real-time human pose recognition in parts from single depth images,” in *2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011, pp. 1297–1304.

[20] Lu Xia, Chia-Chih Chen, and J. K. Aggarwal, “Human detection using depth information by Kinect,” in *2011 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2011, pp. 15–22.

[21] H. Jhuang, E. Garrote, X. Yu, and et al, “Automated home-cage behavioural phenotyping of mice,” *Nat Commun*, vol. 1, 2010.

[22] J. P. Monteiro, H. P. Oliveira, P. Aguiar, and J. S. Cardoso, “Depth-map Images for Automatic Mice Behavior Recognition,” presented at the 1st PhD. Students Conference in Electrical and Computer Engineering, Porto, Portugal, 2012.

[23] L. H. Tecott and E. J. Nestler, “Neurobehavioral assessment in the information age,” *Nat Neurosci*, vol. 7, no. 5, pp. 462–466, 2004.

[24] S. C. Stanford, “The open field test: reinventing the wheel,” *Journal of Psychopharmacology*, vol. 21, no. 2, p. 134, 2007.

[25] J. Archer, “Tests for emotionality in rats and mice: A review,” *Animal Behaviour*, vol. 21, pp. 205–235, May 1973.

[26] P. Curzon, M. Zhang, R. Radek, and G. Fox, “The Behavioral Assessment of Sensorimotor Processes in the Mouse: Acoustic Startle, Sensory Gating, Locomotor Activity, Rotarod, and Beam Walking,” in *Methods of Behavior Analysis in Neuroscience*, 2nd ed., vol. Chapter 8, Boca Raton (FL): CRC Press, 2009.

[27] H. Masuya, S. Yoshikawa, N. Heida, T. Toyoda, S. Wakana, and T. Shiroishi, “Phenosite: a web database integrating the mouse phenotyping platform and the experimental procedures in mice,” *J Bioinform Comput Biol*, vol. 5, no. 6, pp. 1173–1191, Dec. 2007.

[28] L. De Visser, R. Van Den Bos, W. W. Kuurman, M. J. H. Kas, and B. M. Spruijt, “Novel approach to the behavioural characterization of inbred mice: automated home cage observations,” *Genes, Brain and Behavior*, vol. 5, no. 6, pp. 458–466, Aug. 2006.

[29] X. Xue, “Video-Based Animal Behavior Analysis,” Ph.D., The University of Utah, United States -- Utah, 2009.

[30] Y. Liang, V. Kobla, X. Bai, and Y. Zhang, “Unified system and method for animal behavior characterization in home cages using video analysis,” U.S. Patent 7,209,58824-Apr-2007.

[31] P. Crook, T. Lukins, J. Heward, and J. Armstrong, “Identifying semi-invariant features on mouse contours,” in *19th British Mach. Vis. Conf*, 2008.

[32] H. Fröhlich, A. Hoenselaar, J. Eichner, H. Rosenbrock, G. Birk, and A. Zell, “Automated classification of the behavior of rats in the forced swimming test with support vector machines,” *Neural Netw*, vol. 21, no. 1, pp. 92–101, Jan. 2008.

[33] N. Edelman, “Automated Phenotyping of Mouse Social Behavior,” Master thesis, Massachusetts Institute of Technology, Massachusetts, 2011.

[34] P. Aguiar, L. Mendonça, and V. Galhardo, “OpenControl: A free opensource software for video tracking and automated control of behavioral mazes,” *J. Neurosci. Methods*, vol. 166, no. 1, pp. 66–72, 2007.

[35] L. Noldus, A. Spink, and R. Tegelenbosch, “EthoVision: A versatile video tracking system for automation of behavioral experiments,” *Behavior Research Methods*, vol. 33, no. 3, pp. 398–414, 2001.

[36] Noldus Information Technology, “EthoVision XT, Video Tracking software and systems.” [Online]. Available: http://www.noldus.com/animal-behavior-research/products/ethovision-xt. [Accessed: 17-Jan-2012].

[37] J. Pham, S. M. Cabrera, C. Sanchis-Segura, and M. A. Wood, “Automated scoring of fear-related behavior using EthoVision software,” *Journal of Neuroscience Methods*, vol. 178, no. 2, pp. 323–326, Apr. 2009.

[38] San Diego Instruments, Inc., “SMART Video Tracking System.” [Online]. Available: http://www.sandiegoinstruments.com/smart-video-tracking-software/. [Accessed: 17-Jan-2012].

[39] J. J. Crowley, M. D. Jones, O. F. O'Leary, and I. Lucki, "Automated tests for measuring the effects of antidepressants in mice," *Pharmacology Biochemistry and Behavior*, vol. 78, no. 2, pp. 269–274, Jun. 2004.
[40] Stoelting Co., "ANY-maze - Flexible video tracking for neuroscience experiments." [Online]. Available: http://www.anymaze.com/. [Accessed: 17-Jan-2012].
[41] W. R. Jung, H. G. Kim, and K. L. Kim, "Ganglioside GQ1b improves spatial learning and memory of rats as measured by the Y-maze and the Morris water maze tests," *Neuroscience letters*, vol. 439, no. 2, pp. 220–225, 2008.
[42] Qubit Systems Inc., "Video Tracking Software." [Online]. Available: http://qubitsystems.com/animal-and-insect/behaviour-a-i/dv1-video-tracking-software/. [Accessed: 17-Jan-2012].
[43] BIOBSERVE GmbH., "Trackit system family." [Online]. Available: http://www.biobserve.com/products/trackit/index.html. [Accessed: 17-Jan-2012].
[44] P. Graham, K. Fauria, and T. S. Collett, "The influence of beacon-aiming on the routes of wood ants," *Journal of Experimental Biology*, vol. 206, no. 3, pp. 535 –541, Feb. 2003.
[45] Vicon, "Vicon Motus." [Online]. Available: http://www.vicon.com/products/motus.html. [Accessed: 17-Jan-2012].
[46] Actimetrics, "Big Brother - video-based activity monitor." [Online]. Available: http://www.actimetrics.com/BigBrother/. [Accessed: 17-Jan-2012].
[47] J. V. Roughan, S. L. Wright-Williams, and P. A. Flecknell, "Automated analysis of postoperative behaviour: assessment of HomeCageScan as a novel method to rapidly identify pain and analgesic effects in mice," *Lab Anim*, vol. 43, no. 1, pp. 17–26, Jan. 2009.
[48] CleverSys, Inc., "HomeCageScan." [Online]. Available: http://www.cleversysinc.com/products/software/homecagescan/. [Accessed: 26-Apr-2012].
[49] X. Burgos-Artizzu, P. Dollár, D. Lin, D. Anderson, and P. Perona, "Social behavior recognition in continuous video," presented at the Computer Vision and Pattern Recognition 2012, Providence, Rhode Island, 2012.
[50] D. Herrera C, J. Kannala, and J. Heikkila, "Joint Depth and Color Camera Calibration with Distortion Correction," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. PP, no. 99, p. 1, 2012.
[51] D. Herrera C., J. Kannala, and J. Heikkilä, "Accurate and Practical Calibration of a Depth and Color Camera Pair," in *Conference on Computer Analysis of Images and Patterns*, 2011, vol. 6855, pp. 437–445.
[52] Z. Zhang, "A flexible new technique for camera calibration," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 22, no. 11, pp. 1330 – 1334, Nov. 2000.
[53] D. Herrera, "Kinect Calibration Toolbox." [Online]. Available: http://www.ee.oulu.fi/~dherrera/kinect/. [Accessed: 29-Jun-2012].
[54] N. Burrus, "Kinect RGBDemo," *Manctl Labs*. [Online]. Available: http://labs.manctl.com/rgbdemo/. [Accessed: 25-Jun-2012].
[55] J. George, J. Minard, and A. Porter, "RGBDToolkit," *STUDIO for Creative Inquiry at Carnegie Mellon University*. [Online]. Available: http://rgbdtoolkit.com/. [Accessed: 25-Jun-2012].
[56] OpenNI, "OpenNI organization." [Online]. Available: http://openni.org/. [Accessed: 24-Jan-2012].
[57] B. Hellwig, D. Uytvanck, M. Hulsbosch, and A. Somasundaram, "ELAN - Linguistic Annotator version 4.1.2." 2011.
[58] R. C. González and R. E. Woods, "Image Segmentation," in *Digital image processing*, 2nd ed., Prentice Hall, 2002, pp. 567–635.
[59] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans on Syst., Man and Cybernet.*, vol. 9, no. 1, pp. 62–66, 1979.
[60] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *IEEE Conference on Computer Vision and Pattern Recognition*, 1999, vol. 2, pp. 637–663.

[61] Jun Zhang and Jinglu Hu, “Image Segmentation Based on 2D Otsu Method with Histogram Analysis,” in *2008 International Conference on Computer Science and Software Engineering*, 2008, vol. 6, pp. 105–108.
[62] H. Lee and R.-H. Park, “Comments on "An optimal multiple threshold scheme for image segmentation,” *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 20, no. 3, pp. 741 –742, Jun. 1990.
[63] H. Oliveira, J. Cardoso, A. Magalhães, and et al, “Simultaneous Detection of Prominent Points On Breast Cancer Conservative Treatment Images,” in *19th IEEE Conference on Image Processing*, 2012.
[64] M. Piccardi, “Background subtraction techniques: a review,” in *2004 IEEE International Conference on Systems, Man and Cybernetics*, 2004, vol. 4, pp. 3099– 3104 vol.4.
[65] I. MathWorks, *Image Processing Toolbox for Use with MATLAB: User’s Guide*. MathWorks, Incorporated, 2001.
[66] S. B. Kotsiantis, “Supervised Machine Learning: A Review of Classification Techniques,” in *Proceedings of the 2007 conference on Emerging Artificial Intelligence Applications in Computer Engineering: Real Word AI Systems with Applications in eHealth, HCI, Information Retrieval and Pervasive Technologies*, Amsterdam, The Netherlands, The Netherlands, 2007, pp. 3–24.
[67] S. K. Murthy, “Automatic Construction of Decision Trees from Data: A Multi-Disciplinary Survey,” *Data Mining and Knowledge Discovery*, vol. 2, no. 4, pp. 345–389, 1998.
[68] B. C. Lovell and C. J. Walder, “Support Vector Machines for Business Applications,” in *Business Applications and Computational Intelligence*, Hershey, PA., U.S.A.: Idea Group, 2006, pp. 267–290.
[69] M. Pal and P. Mather, “Decision Tree Based Classification of Remotely Sensed Data,” presented at the centre for remote imaging, sensing and processing (CRISP), 2001.
[70] D. P. Huttenlocher, G. A. Klanderman, and W. J. Rucklidge, “Comparing images using the Hausdorff distance,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 15, no. 9, pp. 850 –863, Sep. 1993.
[71] W.-C. Chiu, U. Blanke, and M. Fritz, “Improving the Kinect by Cross-Modal Stereo,” 2011, pp. 116.1–116.10.
[72] J. Kramer, M. Parker, N. Burrus, and F. Echtler, *Hacking the Kinect*. Apress, 2012.
[73] P. Alimi, “Object persistence in 3D for home robotics.” The University Of British Columbia, Apr-2012.
[74] D. A. Butler, S. Izadi, O. Hilliges, D. Molyneaux, S. Hodges, and D. Kim, “Shake’n’sense: reducing interference for overlapping structured light depth cameras,” in *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems*, New York, NY, USA, 2012, pp. 1933–1936.
[75] S. Izadi, D. Kim, O. Hilliges, D. Molyneaux, R. Newcombe, P. Kohli, J. Shotton, S. Hodges, D. Freeman, A. Davison, and A. Fitzgibbon, “KinectFusion: real-time 3D reconstruction and interaction using a moving depth camera,” in *Proceedings of the 24th annual ACM symposium on User interface software and technology*, New York, NY, USA, 2011, pp. 559–568.
[76] Y. Nie, T. Takaki, I. Ishii, and H. Matsuda, “Behavior recognition in laboratory mice using HFR video analysis,” in *2011 IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 1595–1600.
[77] V. N. Vapnik, *The Nature of Statistical Learning Theory*. Springer, 1995.
[78] T. Joachims, T. Finley, and C.-N. Yu, “Cutting-plane training of structural SVMs,” *Machine Learning*, vol. 77, no. 1, pp. 27–59, 2009.
[79] P. Henry, M. Krainin, E. Herbst, X. Ren, and D. Fox, “Rgbd mapping: Using depth cameras for dense 3d modeling of indoor environments,” *IN RGB-D: ADVANCED REASONING WITH DEPTH CAMERAS WORKSHOP IN CONJUNCTION WITH RSS*, 2010.
[80] PrimeSense Ltd., “PrimeSense Natural Interaction.” [Online]. Available: http://www.primesense.com/. [Accessed: 24-Jan-2012].
[81] PrimeSense Ltd., “PrimeSensorTM, Reference Design 1.08.” 2010.
[82] T. Kühn, “Presentation: The Kinect Sensor Platform,” *Technische Universität München ADVANCES IN MEDIA TECHNOLOGY*, no. Media Technology Seminars, 2011.

[83] N. Villaroman, D. Rowe, and B. Swan, “Teaching natural user interaction using OpenNI and the Microsoft Kinect sensor,” in *Proceedings of the 2011 conference on Information technology education*, New York, NY, USA, 2011, pp. 227–232.
[84] Adafruit Industries, “Adafruit Industries, Unique & fun DIY electronics and kits.” [Online]. Available: http://adafruit.com/. [Accessed: 17-Jan-2012].
[85] OpenKinect, “OpenKinect Community portal,” *OpenKinect Main Page*. [Online]. Available: openkinect.org. [Accessed: 17-Jan-2012].
[86] Microsoft Corporation, “Use the Power of Kinect for Windows to Change the World - Kinect for Windows Blog - Site Home - MSDN Blogs,” *Kinect for Windows Blog - MSDN Blogs*, 09-Jan-2012. [Online]. Available: http://blogs.msdn.com/b/kinectforwindows/archive/2012/01/09/kinect-for-windows-commercial-program-announced.aspx. [Accessed: 17-Jan-2012].
[87] Microsoft Corporation, “Kinect for Windows,” *Microsoft Kinect SDK for Developers*. [Online]. Available: http://www.microsoft.com/en-us/kinectforwindows/. [Accessed: 17-Jan-2012].
[88] A. Kar, “Report: Skeletal Tracking using Microsoft Kinect,” Indian Institute of Technology Kanpur, Uttar Pradesh, 2011.
[89] J. Kjaer, “Bachelor Thesis: A Qualitative Analysis of Two Automated Registration Algorithms In a Real World Scenario Using Point Clouds from the Kinect,” Danmarks Tekniske Universitet, Copenhagen, 2011.

References

Appendix A

Depth Sensor Information

Most 3D mapping systems contain three main components: first, the spatial alignment of consecutive data frames; second, the detection of loop closures; third, the globally consistent alignment of the complete data sequence. While 3D point clouds are extremely well suited for frame-to-frame alignment and for dense 3D reconstruction, they ignore valuable information contained in images. However, it is extremely hard to extract dense depth from camera data alone [79].

Recently introduced at large scale in the market, essentially driven by computer gaming and home entertainment applications, RGB-D cameras are sensing systems that capture RGB images along with per-pixel depth information. RGB-D cameras are an emerging trend of technologies that provide high quality synchronized depth and color data. Using active sensing techniques, robust depth estimation has been achieved at real time.

RGB-D cameras rely on either active stereo([15],[16]) or time-of-flight sensing ([17]) to generate depth estimates at a large number of pixels. While sensor systems with these capabilities have been custom-built for years, only now are they being packaged in form factors that make them attractive for research outside specialized computer vision groups. In fact, the key drivers for the most recent RGB-D camera systems are, thus establishing the consumer depth cameras market. RGB-D cameras allow the capture of reasonably accurate mid-resolution depth and appearance information at high (30 fps) data rates.

A.1 The Microsoft Kinect sensor platform

Microsoft's *Kinect*™ [15] was originally developed to control games on the *Xbox 360*™ without having the need of holding a device like a controller. Released in early November 2010 Kinect has demonstrated to effectively and accurately track human motion. Its characteristics led to its widespread popularity and adoption in areas such as robotics or health-care [63].

A.1.1 Hardware

The Kinect is based on a sensor design developed by *PrimeSense Ltd* [80]. The Kinect consists of three optical components: a laser based near infrared (IR) projector, an IR camera (PS1080 CMOS image sensor) and a color camera. For the audio part there is a multiarray microphone, which consists of four sensors and is able to separate sound from different directions.

From the reference design [81] of the PrimeSense sensor is possible to take some technical specifications of the Light Coding ™ that is the technology that allows Kinect to construct 3D depth maps of a scene in real-time. Structured near-infrared light is projected on a region of space and a standard CMOS image sensor is used to receive the reflected light.

The projected speckle pattern repeats itself after 211 horizontal spots and 165 vertical spots and in every of these blocks there is a bright center point. The total pattern is composed of a 3x3 repetition of the before mentioned spots, which results in 633x495 spots. This pattern provides a rich source of easily extracted features. The variation of these features compared against the known pattern for a fixed distance provides a method for depth reconstruction [82]. This represents a computationally less demanding solution compared to the more usual use of two cameras for stereo vision.

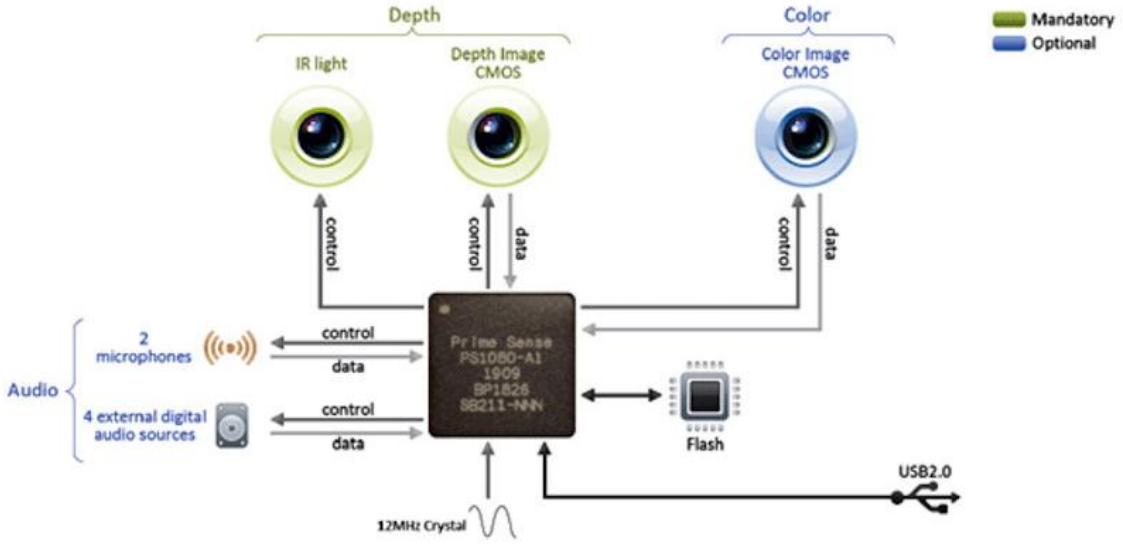

Figure A.1 - Primesensor reference design. From [81].

The depth image size from the PS1080 has a maximum resolution of 640 x 480. At 2m from the sensor it is able to resolve down to 3mm for height and width and 1cm for depth. It operates at a range between 0.8m and 3.5m. Experimentation has shown that Kinect is only able to process depth data at a frame rate of 30 fps. The sensor also has an integrated RGB camera with a maximum resolution of 1600x1200 (UXGA) to match the depth data with real images [83].

A.1.2 Software

Kinect itself actually does not calculate the depth, but returns a more abstract value for the host system to handle. There are several projects with freely available libraries and drivers that can be used to collect and process data from a Kinect sensor.

Even before the official release of Microsoft Kinect on November 4th, 2010 *Adafruit Industries*[84] announced a contest to produce open-source drivers for the Kinect [83]. In response to that call, Hector Martin released his code on November 10th, which marks the beginning of *OpenKinect* community[85] that continued to evolve ever since. The *OpenKinect* community releases the Kinect driver called *libfreenect* (released under Apache 2.0 or GPLv2 license) to connect and use the Kinect with a computer instead of its designated use with the Xbox.

In addition, a group of companies which included *PrimeSense Ltd.*, who has developed the reference design in which the Kinect is based, launched *OpenNI* [56] as a not-for-profit organization that aims to set an industry standard framework for the interoperability of natural interaction devices. With the framework they have released they began to supply the *OpenNI* driver and the *NITE* (Natural Interaction Technology for End-user) Middleware software for scene perception from a compliant device to application-ready data. These libraries are available for applications written in C, C++, and C#. Because Kinect is an *OpenNI*-compliant device, this software can be used to build applications for it. While it has to be noted that the Kinect sensor is not the only device that uses the *PrimeSense* reference design. *ASUS Xtion Pro*[16] is another example of a device that shares the aforementioned design.

Due to the success of the Kinect, Microsoft released an software development kit (SDK) beta for non-commercial purposes only to access Kinect's capabilities on a PC in June 2011. Recently, as Craig Eisler, general manager of Microsoft Kinect for Windows team, announced [86] was launched not only a new Kinect hardware especially for Windows but also the Kinect for Windows commercial program [87]. While presenting some new improved features, such as Kinect developer suite and near mode, the difference of this new Kinect device with the existing equipment can generally be considered negligible or nonexistent.

Both *libfreenect* and OpenNI projects work on Windows, Linux (Ubuntu), and Mac OS X while Microsoft sdk is for windows only. All allow you to access color (8-bit RGB) and depth images (11 bit) in 640x480 resolution at 30 fps [88]. The projects are not compatible and they cannot be used simultaneously. Differences between the libraries are motor control, access/use of the Kinect's image and depth registration, integration with middleware for higher-level, and calibration requirements.

A note here to the fact that although a 640x480 resolution gives a theoretical upper limit of 640x480 = 307200 points (11-bit values) in a point cloud, in practice, a scene with good capturing conditions will result in a cloud of at most 265000 points. The main reason for that discrepancy should be due to how the depth map will map onto the color image, which is captured with a wider field of view. The quality of color images captured by the Kinect is about as good as a decent webcam, and Bayer noise is noticeable [89].

For the purposes of this dissertation, OpenNI was selected because it could be used in multiple operating systems, it is designed and maintained by its member companies to be an industry standard and documentation of the framework at the time of this writing surpasses that of the others

Appendix B

Working Dataset for Classification

In this section some details are presented for the database used for classification method benchmark. Videos were annotated by one scorer using the open source video annotation tool ELAN [57], and validated thereafter. Distribution of behavior labels on the set of 4 videos is presented in Figure B.1. Figure B.2 to Figure B.5 present annotations for two mice behaving differently used for these experiments recorded at Instituto de Biologia Molecular e Celular facilities.

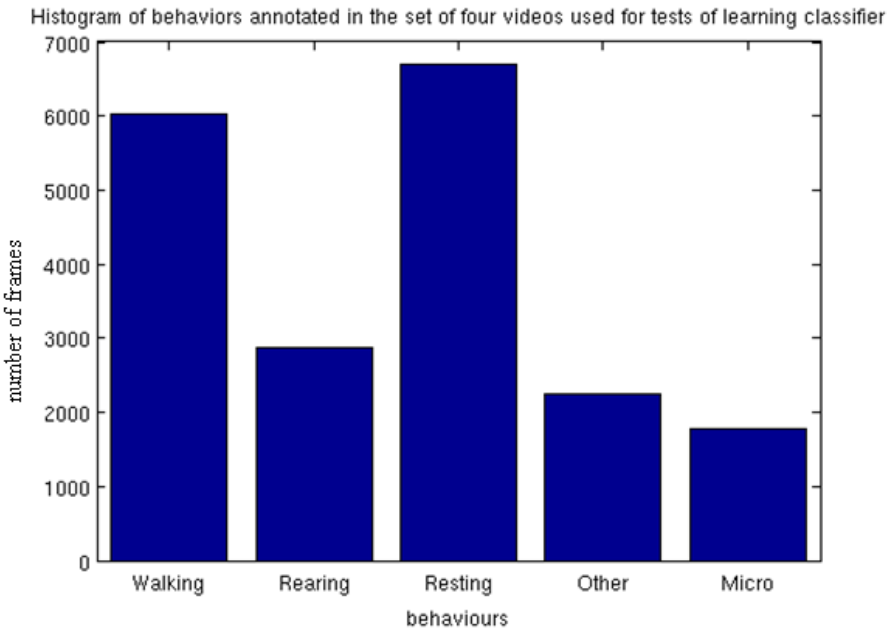

Figure B.1 - Histogram of behaviours in the full set of 4 videos (around 25 minutes) used for classification studies.

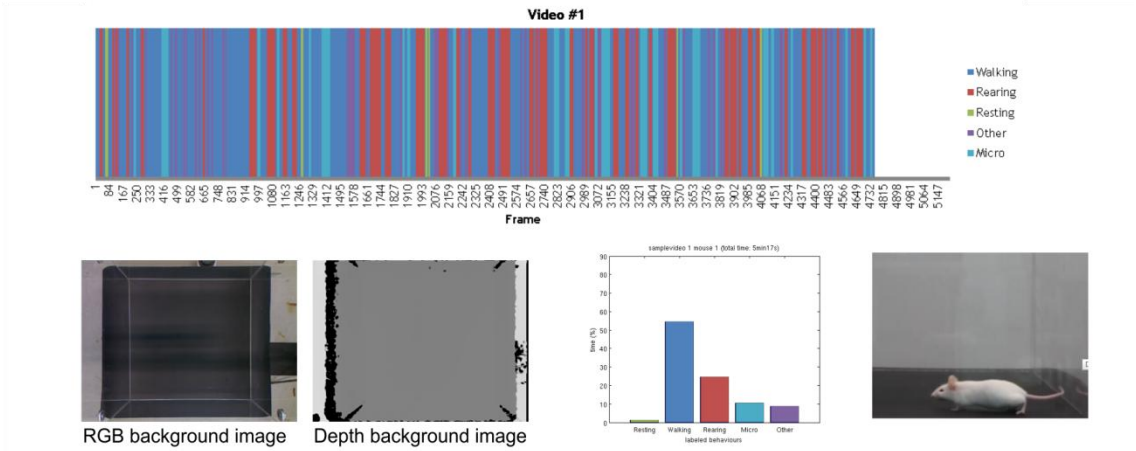

Figure B.2 - Manual annotations performed for video 1 considering the behaviours of walking, rearing, resting, micro movement and other; Behaviors distribution and background images.

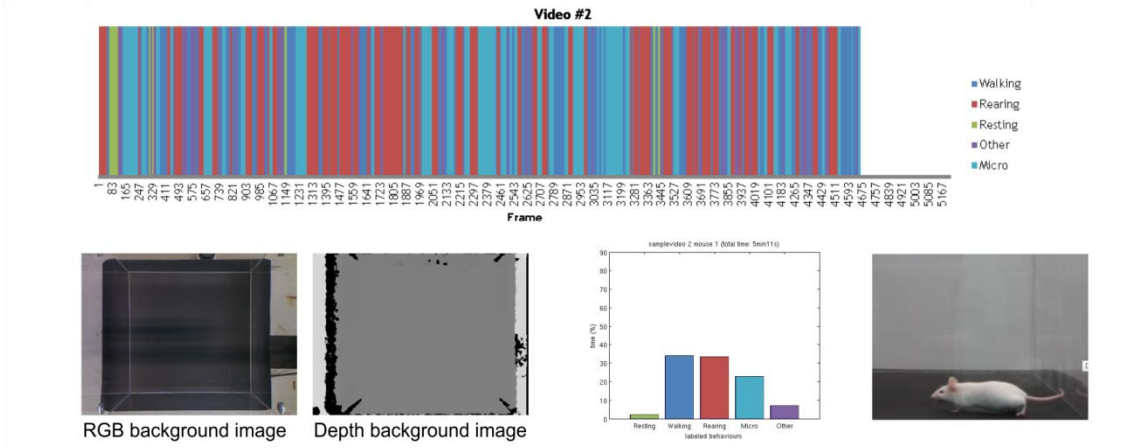

Figure B.3 - Manual annotations performed for video 2 considering the behaviours of walking, rearing, resting, micro movement and other; Behaviors distribution and background images.

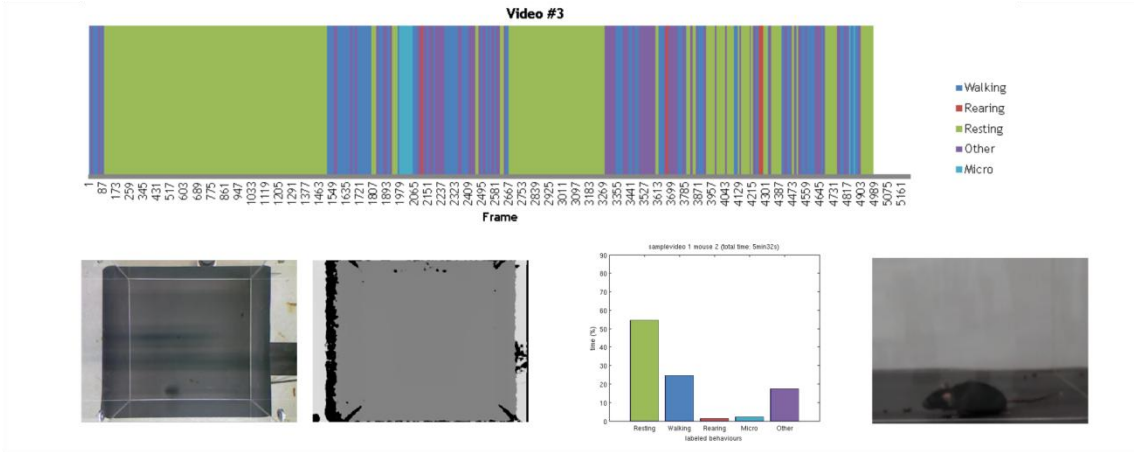

Figure B.4 - Manual annotations performed for video 1 considering the behaviours of walking, rearing, resting, micro movement and other; Behaviors distribution and background images.

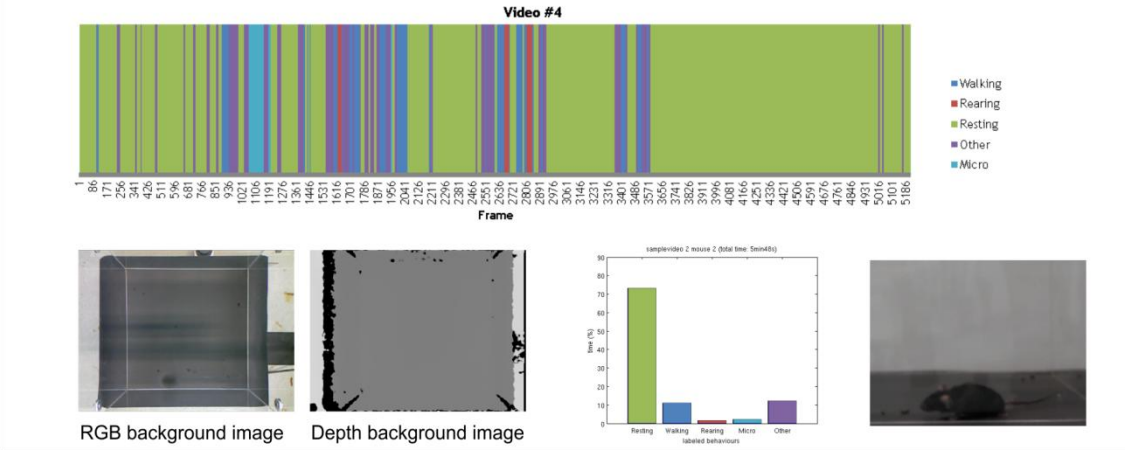

Figure B.5 - Manual annotations performed for video 1 considering the behaviours of walking, rearing, resting, micro movement and other; Behaviors distribution and background images.