



Universidade Católica Dom Bosco
Centro de Ciências Exatas e Tecnológicas
Curso de Engenharia de Computação

Extração de Características Comportamentais de
Camundongos em Ambientes Controlados

Edy Alberth Kamiya

Prof. Orientador: Dr. Hemerson Pistori

*Relatório Final submetido como um dos requisitos
para a obtenção do grau de Engenheiro de Com-
putação.*

UCDB - Campo Grande - MS - Novembro/2005

Capítulo 1

Agradecimento

Agradeço a Deus por permitir o término deste trabalho, e por não deixar a energia cair em momentos críticos. Agradeço aos meus pais, Heitor e Luisa pelo suporte e pela paciência gasta com meu mau humor. A minha irmã Kellen pela preocupação e, por entender que meu trabalho é mais importante que o Orkut dela.

Ao meu orientador Hemerson que guiou meus passos, sempre fazendo mais que sua posição exigia. Obrigado por tentar me transformar em um ser humano melhor! Á professora Priscila “Preta” pela sutiliza no modo de correção deste trabalho.

Agradeço também aos meus amigos “Kanezaki”, “Tata”, “Beah”, “Emerson Mudim”, “Ana Pan”, “Douglas Mise”, por me fazerem rir quando a desgraça era eminente. À nervosa família Ortega, Anita “Tótó”, “Babi”, “Krol”, “Dohda”, “Mari” e Fabio “Nicaum” pela amizade e pelas horas de conversa sem sentido. Aos amigos Fábio Mamoré, Ana Carol, Lw e Ana Elisa pelos “papos-cabeça”, que nem sempre se resumiam em risos.

Aos meus colegas e professores da faculdade por me agüentarem por 5 longos anos. Um agradecimento especial ao calouro Kleber, que forneceu papéis importantes na pesquisa, à Flaviane, que salvou minha pele na entrega da versão parcial deste trabalho, ao Melancia pela festa, e ao Daniel pela “experiência em programação”. Aos três mosqueteiros do projeto Topolino, “Bruno, Broca e Wesley PA”, e ao “Paragua” pelas conversas técnicas que mais ninguém tem paciência de escutar.

E por fim, sou grato a Nova, a MCM e a Odisséia. As três máquinas da minha vida.

Resumo

Os sistemas computadorizados de rastreamento e análise comportamental são amplamente procurados por laboratórios farmacêuticos. Estas ferramentas permitem que os pesquisadores estudem o comportamento de animais por longos períodos de tempo, além disto, fornecem precisão para dados quantitativos e reduzem o custo humano e financeiro em pesquisas. Infelizmente o alto preço destes sistemas demanda investimentos pesados em tecnologia, o que nem sempre é possível para os laboratórios. Este trabalho apresenta um estudo sobre a utilização de técnicas de visão computacional, momentos da imagem e aprendizagem de máquina na construção de um módulo, que realize a extração de características comportamentais de camundongos, utilizando equipamentos de baixo custo e *softwares* livres, reduzindo o preço do sistema.

Abstract

The computerized tracking and behaviour analysis system are widely required by pharmaceutical laboratories. These tools allow researchers to study the animal behaviour for long periods of time. Besides this, they provide precision for quantitative data and reduce the human and financial cost in researchs. Unfortunately, the high price of these systems demands heavy investment in technology, wich is not always possible for the laboratories. This project presents a study about the use of computer vision technics, image moments and machine learn in a module construction. This module realizes the extraction of mice behavior characteristics, using low cost equipments and free softwares, wich reduce the system price.

Conteúdo

1	Agradecimento	2
2	Introdução	9
3	Revisão Bibliográfica	12
3.1	Fundamentos do Processamento de Imagens Digitais	12
3.1.1	Aquisição de Imagens	13
3.1.2	Imagem, Pixel e Espaço de cores	13
3.1.3	Segmentação	17
3.1.4	Limiarização	17
3.1.5	Morfologia Matemática	19
3.2	Aprendizagem de Máquina	23
3.3	Revisão de Conceitos da Cinemática	26
3.3.1	Medidas de distância	26
3.3.2	Velocidade	28
3.3.3	Aceleração	28
3.4	Trabalhos Correlatos	29
4	Momentos Estatísticos	30
4.1	Momentos	30
4.2	Momentos Geométricos	32
4.2.1	Momentos de ordem zero	32
4.2.2	Momentos de primeira ordem	32
4.2.3	Momentos de Segunda Ordem	33
4.3	Momentos centrais	33
4.4	Momentos invariáveis	34
5	Ferramentas	35
5.1	ImageJ	35
5.2	JMF	36
5.3	Weka	36

5.3.1	Arquivo Arff	37
6	Implementação das técnicas	38
6.1	Aquisição da imagens	38
6.2	Extração de características dinâmicas	42
6.2.1	Distância Percorrida	43
6.2.2	Velocidade	44
6.2.3	Aceleração	44
6.2.4	Tempo de permanência do camundongo em uma região	45
6.3	Extração de características posturais	46
6.3.1	Captura das Imagem de Treinamento e Armazenamento	46
6.3.2	Extração dos Momentos Invariantes e Geração do ar- quivo ARFF	49
6.3.3	O classificador	52
7	Experimentos, Resultados e Análise.	53
7.1	Experimentos	53
7.2	Resultados	54
7.3	Análise	57
8	Considerações finais	59
	Referências Bibliográficas	61

Lista de Figuras

3.1	Imagem digital com resolução 8 x 8	14
3.2	Imagem digital com resolução 16 x 16	15
3.3	Pixel p e seus vizinhos	15
3.4	(a)Vizinhança-de-4 (b)Vizinhança-de-8	16
3.5	(a)Imagem colorida (b)Imagem em tons de cinza	16
3.6	Histograma da Figura 2.5(b)	18
3.7	Imagem binária da figura 2.5	19
3.8	Erosão	20
3.9	Dilatação	20
3.10	Abertura	21
3.11	Fechamento	22
3.12	Fechamento da Figura 2.7	23
3.13	Exemplo Rotulado	24
3.14	Fase Aprendizagem	25
3.15	Fase Classificação	25
3.16	Sub-área de uma imagem	27
4.1	Operações as quais os momentos invariantes permanecem constantes à original	31
6.1	Captura de imagens	38
6.2	Imagem capturada pela Webcam	39
6.3	Esquema de captura de imagens	39
6.4	Operações na imagem	40
6.5	Operações na máscara	40
6.6	Dilatação e Erosão	41
6.7	Subtração	41
6.8	Centro de massa de um camundongo	42
6.9	Calibragem	43
6.10	Gráfico da Velocidade	44
6.11	Gráfico da Aceleração	44

6.12	Seleção da Área de Observação	45
6.13	Experiência em Andamento	45
6.14	Normal	46
6.15	Encolhido	46
6.16	Esquema da formação do banco de imagens	47
6.17	Rotulador	47
6.18	Estrutura do banco de imagens	48
6.19	Criação de Instâncias	49
6.20	As 4 rotações de uma imagem	50
6.21	Deslocado	50
6.22	Aumentada em 30%	50
6.23	Formação do Arff	51
6.24	Treinamento	52
6.25	Classificador Atuando	52
7.1	Experimento	53
7.2	Características Dinâmicas	54
7.3	Tabela de Regiões	54
7.4	Validação Cruzada	55

Capítulo 2

Introdução

Ratos e camundongos são animais amplamente utilizados em laboratórios [4]. Experiências envolvendo estes roedores são descritas em pesquisas abrangendo doenças como Parkinson [25] e Alzheimer [30]. Superficialmente, é possível dizer que, o uso destes animais tem como objetivo simular o efeito de uma determinada substância em um ser humano.

A conclusão destas experiências é baseada em resultados obtidos durante ou após a sua execução. Existem vários meios de extrair estes dados, por exemplo, o exame de vísceras em busca de anomalias [20] ou análise do comportamento animal [22], o ultimo sendo o foco deste trabalho.

Na análise do comportamento, cada ação do animal se enquadra em uma categoria de comportamento [4]. Desta forma, é possível reconhecer a ansiedade em uma experiência onde um rato prefere permanecer nas bordas ao centro quando abandonado em um campo aberto, cercado por um anteparo em formato de círculo[16].

Uma proposta de categorização destes comportamentos ou domínios comportamentais é descrita por Wehner [34] que os separa em sete grupos: ansiedade e medo, susto e inibição pré-pulso do susto, modelos de depressão, medida de impulsividade em ratos, modelos de aprendizado memória, e medidas de hiper atividade e falta de atenção em ratos.

Estas experiências podem ser empregadas em diversas áreas para elucidar questões ainda não esclarecidas e fornecer conclusões mais sólidas aos pesquisadores. É possível encontrar sua influência na neurociência [11], na farmacologia [6], na toxicologia [22], na psicologia [32] e na genética [35, 19, 17] entre outras ciências.

Considerando a extensa área de atuação da análise do comportamento animal, inúmeras experiências são realizadas e na maioria delas a metodologia é compartilhada. O comportamento é registrado de forma manual ou semi-automática. Neste ultimo caso, são utilizados programas que arquivam

os dados, melhorando a organização [23], mas exige um observador que interprete e registre cada ação do roedor e seus respectivos parâmetros como a hora, o começo e o fim de alguma atividade. O programa MICE é um exemplo de software que ajuda na organização das experiências [13].

Contudo, a tarefa de observar e registrar o comportamento do animal em uma experiência é rotulada como intensiva, subjetiva, e sujeita às tendências do observador, podendo sofrer os efeitos da fadiga e da expectativa que um determinado resultado apareça [4, 23]. Levando-se em conta tais dificuldades, a elaboração de uma solução que resolva ou amenize estes problemas pode levar a qualidade das experiências à níveis superiores, comparados ao atual modelo.

Uma das soluções é a elaboração de um sistema automático de análise comportamental. Ele armazena informações sobre a ação do animal tais como a posição e a velocidade sem a presença de um observador. Este funcionamento automático isenta o sistema das fraquezas do observador, desta maneira é possível realizar experiências que necessitam de longos períodos de tempo de observação do animal. Em outras palavras, experimentos envolvendo grandes períodos de inatividade do roedor ou mudanças do comportamento relacionado com a troca de turnos, diurno e noturno, podem ser realizados sem um grande custo humano devido à automatização do sistema.

O processo de automatização resume-se em um sistema que possui um mecanismo capaz de fornecer dados sobre o ambiente e, um software capaz de interpretar tais informações, sem a necessidade de um supervisor humano. EthoVision, por exemplo, é um sistema composto de uma câmera ou um vídeo cassete, que fornecem as imagens, e um software que analisa essas informações [23].

O objetivo deste trabalho é a implementação de um módulo capaz de extrair e identificar características comportamentais de camundongos. Elas são divididas em 2 grupos, um relacionado com movimentos e a outro com a forma. O primeiro grupo engloba características extraídas com a análise do movimento do camundongo, tais como velocidade, tempo de permanência em uma região, aceleração e distância percorrida. O cálculo destas medidas será realizado através de fórmulas encontradas na física e na geometria. O segundo grupo classifica o comportamento através do formato do camundongo. Determinadas ações como “ficar em pe” ou “se limpar”, modificam a forma do camundongo, permitindo a identificação postural. Em ambos os grupos a base para os cálculos têm como origem os momentos da imagem. Os momentos da imagem são dados estatísticos que revelam informações sobre a imagem, destas informações utilizaremos duas, o centro de massa e os momentos invariantes, explicados posteriormente. O módulo implementado

será integrado ao projeto Topolino.¹.

O projeto Topolino é um sistema de apoio para o estudo do comportamento de animais de laboratório em desenvolvimento pelo Grupo de Pesquisa em Engenharia e Computação (GPEC) e Centro de Ciências Biológicas e da Saúde (CCBS) da Universidade Católica Dom Bosco, UCDB, em parceria com o Laboratório de Linguagens e Técnicas Adaptativas (LTA) da Universidade de São Paulo, USP. Este sistema ajudará o pesquisador na realização de experiências com animais de laboratório. O princípio destes sistemas é a utilização de webcams capazes de monitorar as ações de um animal. Computadores são utilizados para a interpretação das imagens registradas e arquivamento dos dados importantes da experiência. Além disso, bancos de dados organizam a disposição dos arquivos, facilitando a busca e o cruzamento das informações.

No capítulo 2 encontra-se a revisão bibliográfica com a fundamentação teórica, os trabalhos correlatos e as fórmulas utilizadas. No capítulo 3 está contido um estudo sobre os momentos estatísticos. No capítulo 4 foram descritas as ferramentas utilizadas e no capítulo 5 encontra-se a implementação. O capítulo 6 é reservado para a descrição dos experimentos e exposição dos resultados e da análise. As considerações finais estão contidas no capítulo 7.

¹<http://www.glug.ucdb.br/pistori/topolino>

Capítulo 3

Revisão Bibliográfica

Neste capítulo encontra-se a fundamentação teórica necessária para o entendimento das técnicas de visão computacional aplicadas nos capítulos seguintes. Em seguida, serão relatados alguns conceitos sobre aprendizagem de máquina, utilizados para a identificação postural do camundongo. Na seção de trabalhos correlatos segue a descrição de trabalhos, semelhantes a este, e que serviram de inspiração para este projeto. Na seção de revisão de conceitos da cinemática serão demonstradas as fórmulas utilizadas neste trabalho.

3.1 Fundamentos do Processamento de Imagens Digitais

Existem duas áreas de grande atuação do processamento de imagens digitais, a melhoria de informação visual para interpretação humana e o processamento de dados de cenas para percepção automática através das máquinas [9]. Este trabalho tem como foco o desenvolvimento desta segunda área de atuação, considerando que o sistema em si deve funcionar sem a interferência humana. Todo tratamento realizado na imagem tende a melhorar a extração de informações pela máquina. Nas seções a seguir segue a descrição de alguns conceitos sobre o processamento de imagens digitais.

A primeira delas descreve a importância da escolha da aquisição de imagens. Na seção seguinte serão apresentados conceitos sobre os elementos básicos da visão computacional, tais como a imagem e o pixel. Na terceira seção segue um estudo sobre a segmentação, que é um dos processos realizados sobre imagem. Outro processo estudado é o da limiarização, que se encontra na seção seguinte. Por último, um breve estudo sobre a morfologia matemática, que pode suavizar os ruídos causados por estes processos, entre

outras tarefas.

3.1.1 Aquisição de Imagens

A aquisição da imagem é o primeiro passo para o processamento. Existem várias formas de se obter uma imagem, seja ela digitalizando uma fotografia, ou separando um quadro de um filme ou simplesmente tirando uma foto com uma máquina digital CCD ¹. Cada uma destas formas possui seus prós e contras, a escolha do meio de captura pode melhorar ou facilitar o processamento da imagem.

A maioria dos trabalhos encontrados sobre rastreamento de animais utiliza câmeras de vídeo para o monitoramento, [2, 4, 24, 23]. Porém este mecanismo de captura de imagens não se adequaria para o problema proposto por Zurn, Jiang e Motai [38]. Os pesquisadores descrevem a captura de imagens no período noturno, ou seja, na ausência de luz, utilizando câmeras e iluminação infravermelhas.

Anos após a publicação deste primeiro artigo, Zurn e Motai escreveram um trabalho com Hohmann e Dworkin. A publicação era a descrição de um sistema híbrido que funciona tanto durante o dia quanto a noite, ciclo claro e escuro. Em suma, era a proposta de um sistema que utiliza dois métodos de captura, com a câmera normal e com a infravermelha [39].

Infelizmente a maioria das imagens capturadas possui excesso de informações. Neste caso, é necessário separar o objeto de interesse do meio, diminuindo o gasto computacional bem como evitando erros provocados pelos dados irrelevantes. As técnicas de segmentação de imagens são descritas na seção seguinte.

3.1.2 Imagem, Pixel e Espaço de cores

Gonzalez e Woods [9] definem o termo imagem como uma função de intensidade luminosa bidimensional representada por $f(x,y)$ cujo x e y representam a localização. Esta explicação se refere a uma imagem em tons de cinza, explicada posteriormente, que carrega apenas o brilho como atributo. É possível então imaginar a imagem como sendo uma matriz bidimensional de valores contínuos que armazenam os valores cinza para cada coordenada. A imagem formada deste modo será denominada imagem real. Infelizmente por limitações computacionais, é inviável o trabalho com ela, visto que é exigido um espaço ilimitado para registrar todas as coordenadas e todos os tons.

¹Charge Coupled Device

Uma maneira de economizar espaço e possibilitar a manipulação da imagem é estipular um ponto que represente uma determinada área desta imagem real, desta forma a figura inteira pode ser representada por pontos. Cada uma destas partes constituintes da imagem é denominada pixel ou pel. Na união dos pixels, organizados em uma matriz pelas posições da figura original, é obtido uma imagem semelhante à real denominada imagem digital.

O resultado desta operação apresenta perdas na qualidade, felizmente cada problema possui um nível de aceitação destas perdas. Nota-se que o aumento no tamanho da área a ser coberta por um pixel é inversamente proporcional a essa qualidade. Em outras palavras, quanto maior o espaço a ser representado por um elemento, pior a imagem e menor será o número de pontos necessários para a constituição da imagem digital. Essa quantidade de pixels, ou grau de detalhes discerníveis, é denominada resolução [9].

Os efeitos causados pela digitalização das imagens podem ser vistos na Figura 3.1, onde o primeiro quadro apresenta a imagem real ² de um círculo, o segundo demarca a área a ser representada por cada pixel e a última como a figura digital. A área coberta por um pixel era de 1/64 do tamanho da imagem, o resultado foi a deformação do círculo e, conseqüentemente, da imagem.

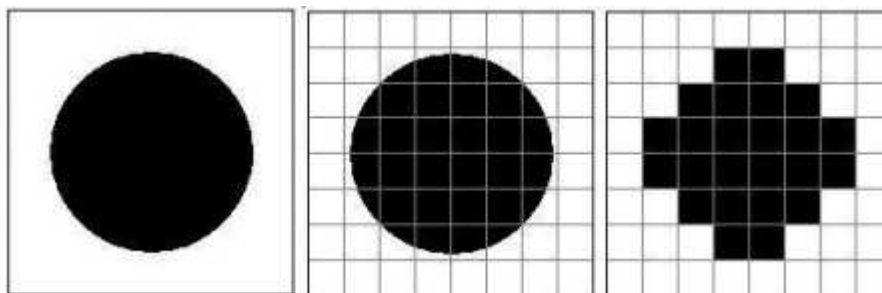


Figura 3.1: Imagem digital com resolução 8 x 8

A Figura 3.2 apresenta a mesma esquematização porém com uma resolução de 16 x 16, sendo o primeiro número a quantidade de linhas e o segundo o de colunas. Esta imagem se aproxima mais da real em relação a Figura 3.1, desta forma observamos que ao diminuir a área representada por cada pixel, há um aumento na resolução, promovendo uma melhora na qualidade da imagem digital.

Cada ponto deve possuir uma coordenada na matriz e atributos que o possibilite caracterizar a área que ele está representando. Todo o pixel p

²Esta imagem digital simula uma imagem real

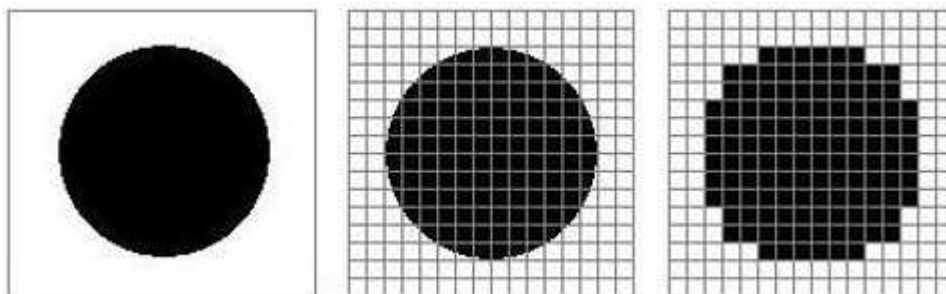


Figura 3.2: Imagem digital com resolução 16 x 16

nas coordenadas (x,y) , em um plano infinito possui vizinhos horizontais e verticais de coordenadas

$$(x + 1, y), (x - 1, y), (x, y + 1), (x, y - 1)$$

e vizinhos diagonais de coordenadas

$$(x + 1, y + 1), (x + 1, y - 1), (x - 1, y + 1), (x - 1, y - 1)$$

Os primeiros representados em branco e os outros em cinza na Figura 3.3.

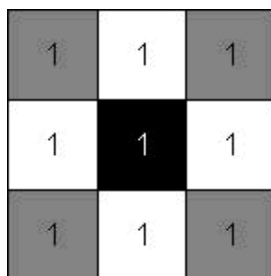


Figura 3.3: Pixel p e seus vizinhos

O conceito de vizinhança é simplesmente a união dos pixels vizinhos. A vizinhança-de-4, apresentada no primeiro quadro da Figura 3.4, é formada pelos pixels vizinhos no sentido vertical e horizontal e a vizinhança-de-8, apresentada no segundo, é a junção da vizinhança-de-4 mais os vizinhos diagonais de p .

As características carregadas por essas unidades podem ser divididas em dois grupos, a crominância e a luminância. A luminância é a componente que contém as informações de brilho, também chamada de tons de cinza. Isto se

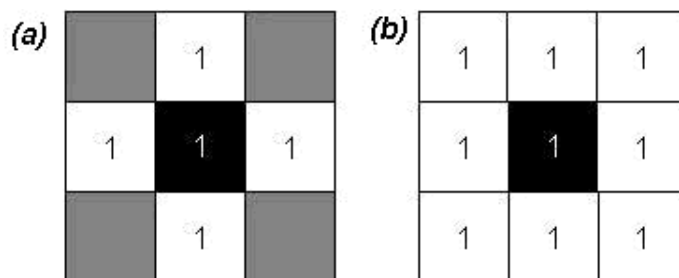


Figura 3.4: (a)Vizinhança-de-4 (b)Vizinhança-de-8

deve ao fato dela indicar a tonalidade, que pode variar do mais claro ao mais escuro, de uma cor.

A crominância é a componente que agrega as cores à uma imagem [8]. Isto é possível graças ao estudo do espaço de cores, que buscam um modelo que represente as cores do mundo real. Existem vários espaços de cores, entre eles o HSV (*Hue*/ Matiz, *Saturation* / Saturação, *Value*/Valor), o CMYK (*Cyan*/ Ciano, *Magenta*/Magenta, *Yellow*/Amarelo, *Black*/Preto), o RGB (*Red*/Vermelho, *Green*/Verde, *Blue*/Azul). A Figura 3.5 (a) demonstra uma imagem no espaço de cores RGB e a Figura 3.5 (b) demonstra uma imagem em tons de cinza.



Figura 3.5: (a)Imagem colorida (b)Imagem em tons de cinza

3.1.3 Segmentação

A segmentação é definida por Gonzalez e Woods [9] como sendo a divisão da imagem em partes ou objetos constituintes. Tal segregação é importante, pois, com a possibilidade de exclusão de informações irrelevantes, como o fundo de uma imagem ou objetos insignificantes no contexto de um problema, há uma diminuição no gasto computacional e ganho em tempo de processamento visto que a área a ser analisada diminui e se restringe, na maioria dos casos encontrados, à região de interesse.

O processo de segmentar a imagem exige a escolha de um ou mais parâmetros que permitam a diferenciação entre os objetos ou regiões da imagem, o sucesso da segmentação está diretamente relacionado com esta escolha que pode ser a forma [28], a textura [33], a textura combinada com a cor [1], a cor [5], entre outras.

A segmentação pela diferença de imagens é uma maneira simples de identificar e segregar um objeto. A comparação de duas imagens pixel a pixel, uma com apenas o fundo da imagem $m(x,y)$ e outra com o objeto sobrepondo o fundo $f(x,y,t_i)$ resultará em apenas o objeto, sendo x,y as coordenadas dos pixels e o t_i como o instante da imagem. A imagem da diferença entre duas imagens pode ser definida como:

$$d_{i,m}(x,y) = \begin{cases} 1 & \text{se } |f(x,y,t_i) - f(x,y,m)| > \Theta \\ 0 & \text{caso contrário} \end{cases}$$

O limiar Θ deve corresponder a 0 trabalhando com imagens binárias, que são formadas apenas por "0 ou 1" ou "preto ou branco". Este tipo de imagem pode ter origem em um processo denominado binarização.

3.1.4 Limiarização

O processo de binarização é a forma mais simples de limiarização que consiste em diferenciar as regiões de uma imagem quando esta apresenta duas classes (o fundo e o objeto). O primeiro passo para binarizar uma imagem é a análise de seu histograma.

O histograma de uma imagem, em tons de cinza, é um gráfico que contém a frequência do aparecimento de cada tom de cinza. Considere uma imagem formada por um objeto e um fundo. Os dois elementos são compostos por pixels carregando tons de cinza diferentes. Pela análise do histograma desta imagem, é possível estabelecer um valor de corte que separe o objeto do fundo.

O histograma da imagem 3.5(b) aparece na Figura 3.6, como pode ser observado os níveis de cinza mais evidentes são os próximos ao branco (255), que representam ao fundo, e os próximos ao cinza (170) representando o objeto.

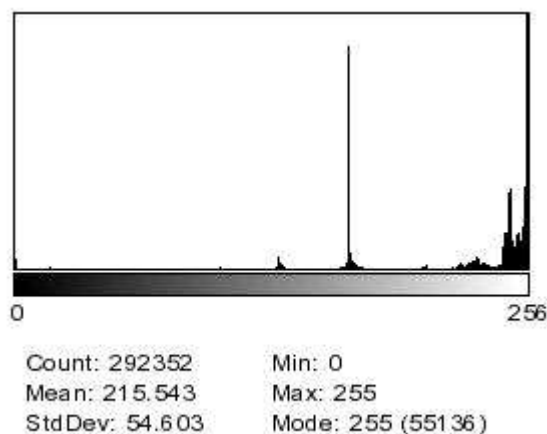


Figura 3.6: Histograma da Figura 2.5(b)

A análise do histograma permite a conclusão que existem duas regiões distintas e que um nível entre o 170 e o 255 poderia segregar as duas regiões. O número 212, a média entre os dois níveis, é escolhido e agora ele será chamado de limiar T . Segundo Gonzalez e Woods [9], para uma imagem $f(x,y)$, a Figura 3.6 neste caso, se transformar em uma imagem limiarizada $g(x,y)$ deve-se seguir essa regra:

$$g(x, y) = \begin{cases} 1 & \text{se } f(x, y) > T \\ 0 & \text{se } f(x, y) \leq T \end{cases}$$

Este processo gerou uma imagem binária conforme a Figura 3.7.



Figura 3.7: Imagem binária da figura 2.5

Repare que na figura binária, dentro do objeto de interesse existem pontos em branco, isto indica a presença de ruído. O ruído é definido como qualquer distúrbio não desejado no sinal [14]. Neste caso, existem pontos dentro do objeto que são confundidos com o meio. Os efeitos causados por estes distúrbios podem ser amenizados com o uso da morfologia matemática.

3.1.5 Morfologia Matemática

A morfologia matemática é uma ferramenta para extração de componentes de uma imagem, podendo ser usada na descrição ou representação de regiões. Ela pode ser usada também no pré e pós - processamento como filtragem morfológica, afinamento e poda. Baseada na teoria dos conjuntos desenvolvida na matemática, ela é composta de simples definições, que combinadas, podem resultar em soluções para os numerosos problemas de processamento de imagem. Porém neste caso os conjuntos serão as regiões de uma figura, ou seja, os pixels. [9] Duas operações utilizadas no processamento de imagens são a dilatação e a erosão. Seus fundamentos e a descrição das definições que as compõem podem ser encontrados em [12].

O efeito do uso dessas operações pode ser visto na Figura 3.8 onde o conjunto A, Figura 3.8 (a), após sofrer uma erosão pelo elemento estruturante B, tem suas proporções reduzidas, Figura 3.8 (b). O efeito é contrário na Figura 3.9, onde o conjunto A, Figura 3.9 (a), após sofrer uma dilatação pelo elemento estruturante B, denotada por $A \oplus B$, tem suas medidas expandidas, Figura 3.9 (b).

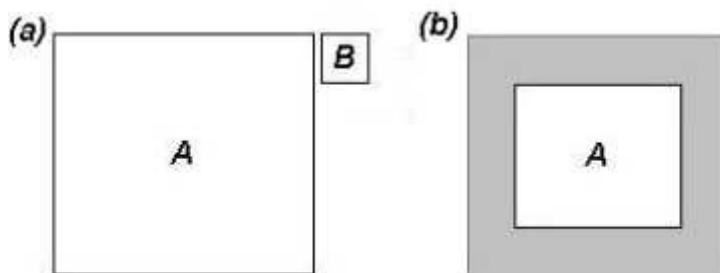


Figura 3.8: Erosão

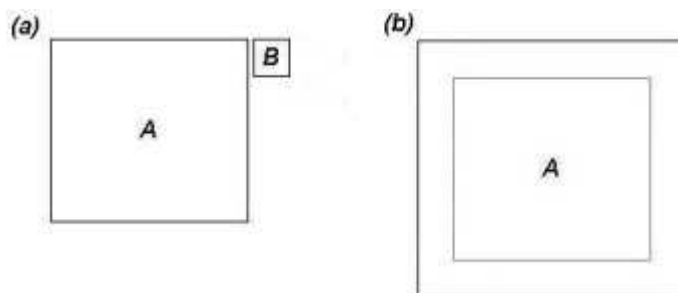


Figura 3.9: Dilatação

A combinação das operações de erosão e dilatação dá origem a duas outras operações, a abertura e o fechamento. A abertura é uma operação que resulta em suavização dos contornos, eliminação de istmos finos e pequenos objetos. Sua denotação é dada por $A \circ B = (A \oplus B) \ominus B$, ou seja a erosão de A por B, dilatada por B, e seus efeitos podem ser observados pela Figura 2.10.

A Figura 3.10 (a) demonstra um conjunto A e o conjunto estruturante B enquanto na Figura 3.10 (b) mostra as várias posições do elemento estruturantes. A Figura 3.10 (c) é o resultado final da erosão, houve a diminuição das medidas de A e a eliminação do istmo. Na Figura 3.10 (d) o conjunto A no processo de dilatação. O resultado final da abertura pode ser visto na Figura 3.10 (e).

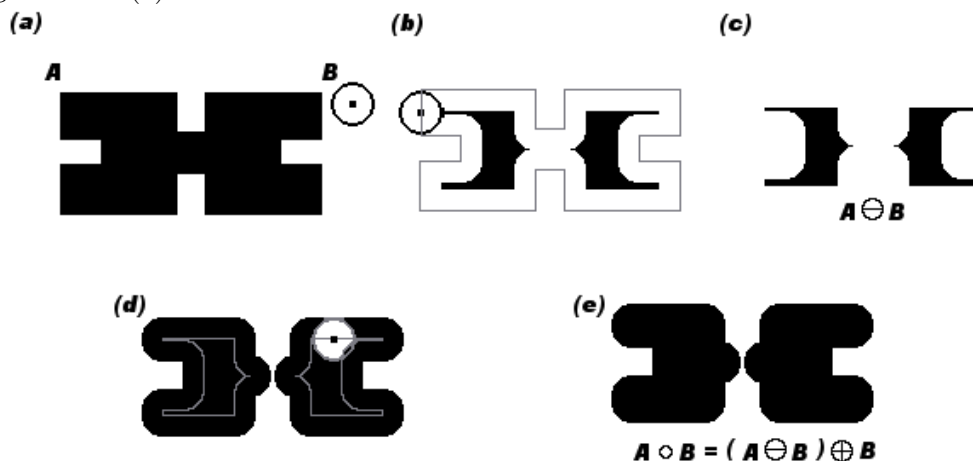


Figura 3.10: Abertura

O fechamento é uma operação que tenta suavizar os contornos, funde pequenas fenda e elimina pequenos buracos dentro dos objetos. Sua denotação é dada por $A \bullet B = (A \ominus B) \oplus B$, ou seja a dilatação de A por B erodida por B , e seus efeitos podem ser observados pela Figura 3.11.

A Figura 3.11 (a) demonstra um conjunto A e o conjunto estruturante B enquanto na Figura 3.11 (b) as medidas são expandidas em relação à original. A Figura 3.11 (c) é o resultado final da dilatação, houve a diminuição da baia que se encontrava a esquerda do conjunto A. A Figura 3.11 (d) é o conjunto A no processo de erosão. O resultado final do fechamento pode ser visto na Figura 3.11 (e).

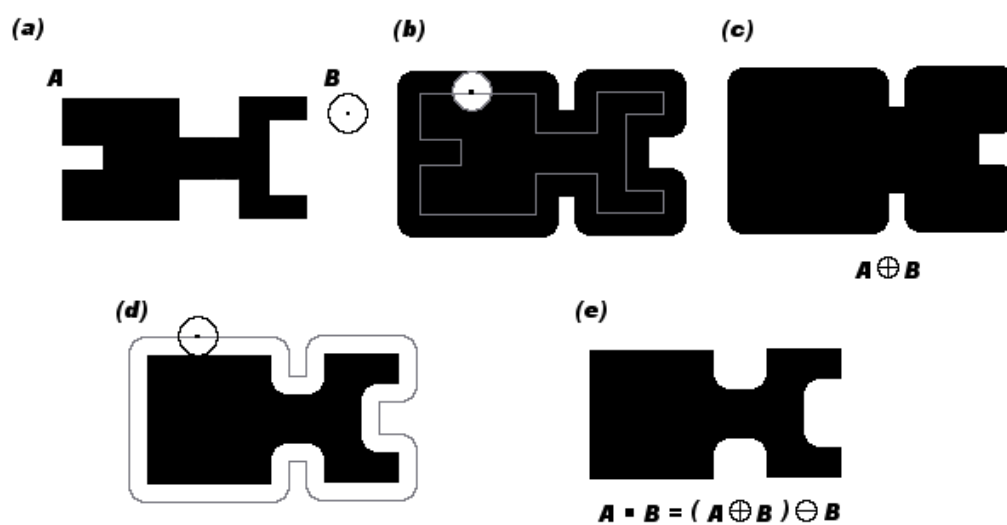


Figura 3.11: Fechamento

Os efeitos da utilização da operação de fechamento para o eliminar os pequenos buracos de um objeto pode ser visto na Figura 3.12.



Figura 3.12: Fechamento da Figura 2.7

3.2 Aprendizagem de Máquina

Segundo Pila [26] existem dois tipos de aquisição de informações, a explícita e a implícita. A primeira envolve a ação de um especialista da área, alguém que organize os dados e forneça ao computador o conhecimento sobre o domínio do problema. A segunda forma de aquisição é um pouco mais complexa, o conhecimento é extraído diretamente dos dados. A responsabilidade da extração deste conhecimento é de uma área da Inteligência Artificial denominada Aprendizagem de Máquina.

A aprendizagem de máquina pesquisa métodos computacionais relacionados à aquisição de novos conhecimentos, novas habilidades e novas formas de organizar o conhecimento já existente. Um sistema de aprendizagem de máquina é um programa de computador que toma decisões baseadas em experiências acumuladas contidas em exemplos previamente resolvidos com sucesso [21].

Técnicas de aprendizagem de máquina podem ser divididas em aprendizagem supervisionada e não supervisionada. A não supervisionada não possui uma classificação prévia dos exemplos de treinamento, ela se preocupa em estabelecer agrupamentos de padrões e identificar possíveis classes. Alguns algoritmos desta categoria são as redes neurais do tipo ma-

pas auto-organizáveis, o algoritmo k-médias e os algoritmos de agrupamento hierárquico.

Na supervisionada o conhecimento é extraído de exemplos já rotulados. Ela utiliza este conhecimento para classificar corretamente futuras amostras não rotuladas. Alguns algoritmos desta categoria são as redes neurais artificiais do tipo multilayer perceptron ³, as máquinas de vetores de suporte, os algoritmos genéticos e as árvores de decisão [31].

Os exemplos, utilizados na aprendizagem de máquina supervisionada, são compostos de características denominadas atributos. Seja n o número de atributos do conjunto $x = (x_1, x_2, x_3, \dots, x_n)$, e a classe ou rótulo associado a cada exemplo, y , a tarefa é encontrar um mapeamento f tal que $y = f(\mathbf{x})$ [26]. A Figura 3.13 é uma representação de um exemplo, ela contém alguns atributos além do atributo especial classe.

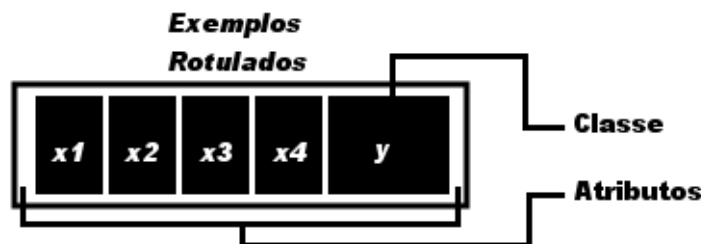


Figura 3.13: Exemplo Rotulado

As classes podem assumir valores contínuos ou categóricos. Quando os exemplos estão rotulados com valores contínuos, a tarefa de aprendizagem é chamada de regressão. Ao se utilizar valores categóricos, a tarefa é chamada de classificação.

³Multicamadas formadas por neurônios perceptrons

Em um sistema de aprendizagem de máquina pode ser dividido em duas fases. Na primeira fase um grupo de exemplos já rotulados é repassado ao sistema. O algoritmo indutor recebe essa entrada e extrai o conhecimento dos atributos. Através deste conhecimento, o sistema gerará um classificador representado em uma estrutura interna (Figura 3.14).

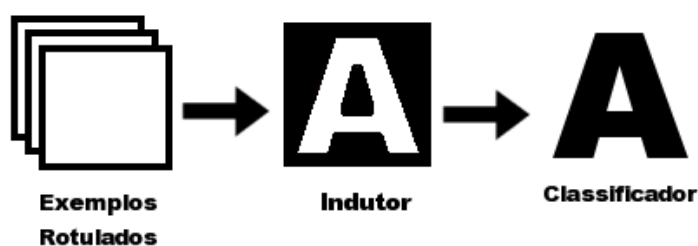


Figura 3.14: Fase Aprendizagem

Na segunda fase, o classificador gerado recebe novos exemplos que não possuem rótulo. A resposta esperada dele é a classificação das entradas (Figura 3.15).

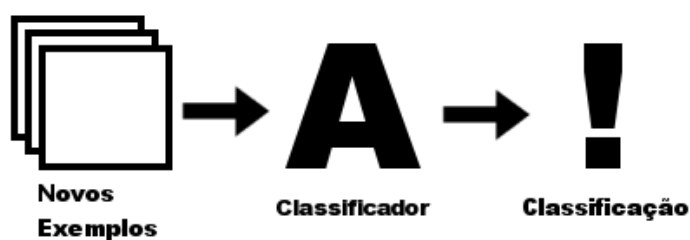


Figura 3.15: Fase Classificação

A precisão do classificador está diretamente relacionada com a qualidade dos atributos dos exemplos de treinamento. Bons atributos são aqueles que são capazes de representar o conhecimento implícito nos dados. Se essa representação não for boa o suficiente, o algoritmo indutor pode gerar um classificador que pode, eventualmente, predizer uma classe errônea a uma determinada entrada.

3.3 Revisão de Conceitos da Cinemática

Nesta seção serão apresentados alguns conceitos sobre as medidas de distâncias, velocidade e aceleração. Eles serão usados na extração de características dinâmicas, que estão relacionadas com o movimento do camundongo.

3.3.1 Medidas de distância

Uma das informações a serem extraídas é o deslocamento efetuado pelo camundongo entre dois instantes. Se as posições espaciais do animal fossem dispostas em um mesmo plano, seria possível o cálculo deste deslocamento. Existem algumas maneiras de estimar a distância entre dois pontos, mas todas elas obedecem aos critérios abaixo.

Em um ambiente de 2 dimensões, dado 3 pontos p , q , z , com coordenadas (x,y) , (s,t) e (u,v) .

D é uma função de distância ou métrica se:

- $D(p, q) \geq 0$ ($D(p, q) = 0$ se e somente se $p = q$). Ou seja, a distância entre dois pontos é ou positiva ou igual a zero se a distância for de um ponto a ele mesmo.
- $D(p, q) = D(q, p)$. A distância entre o ponto p ao q é a mesma que entre o q ao p
- $D(p, z) \leq D(p, q) + D(q, z)$ A distância entre os pontos p e z deve ser menor ou igual a soma da distância entre os pontos p e q , e q e z

Foram encontradas três métricas que poderiam satisfazer o problema, a distância de 4 (conhecida por distância de Manhattan ou *city-block* ou quarteirão), a de 8 (conhecida por distância Xadrez) e a euclidiana. Esta última foi escolhida para ser usada neste projeto seguindo a necessidade do usuário. Ela é baseada no teorema de Pitágoras em um espaço euclidiano contínuo.

As imagens digitais estão em um universo discreto, o uso da distância euclidiana nestas imagens pode resultar em valores não quantizados, por isso, nem sempre ela pode ser utilizada ao se trabalhar com imagens. No trabalho em questão, os valores contínuos são aceitos.

A Figura 3.16 é uma sub-área que contém dois pontos, $p(1,3)$ e $q(3,5)$, o cálculo da distância será realizado pelas três métricas para comparação.

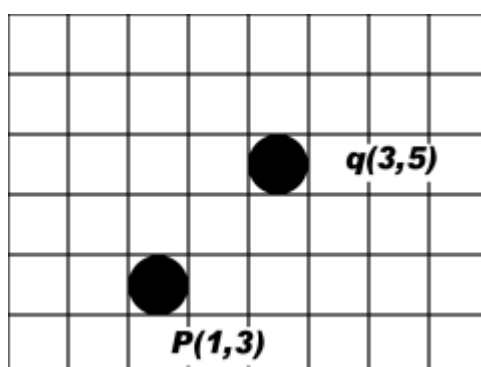


Figura 3.16: Sub-área de uma imagem

A distância de 4 entre p e q é a soma da diferença entre os pontos:

$$D_4(p, q) = |x - s| + |y - t| \quad (3.1)$$

$$D_4(p, q) = |1 - 3| + |3 - 5| = 4$$

A distância de 8 entre p e q é valor da maior diferença:

$$D_8(p, q) = \max(|x - s|, |y - t|) \quad (3.2)$$

$$D_8(p, q) = \max(|1 - 3|, |3 - 5|) = 2$$

A distância Euclidiana entre os pontos p e q é definida como:

$$D_e(p, q) = \sqrt{(x - s)^2 + (y - t)^2} \quad (3.3)$$

$$D_e(p, q) = \sqrt{(1 - 3)^2 + (3 - 5)^2} = 2,8284$$

3.3.2 Velocidade

A velocidade é uma forma de medir o movimento que uma partícula realiza, ou seja, o deslocamento em função do tempo. Ao utilizar o módulo desta medida, encontramos a velocidade escalar. Ela é diferente da velocidade vetorial por não apresentar resultados negativos e nem fornecer informações sobre a direção [10]. Quando se utiliza a distância euclidiana, o deslocamento sempre será positivo, uma vez que ela é o resultado de uma raiz quadrada, e como o tempo nunca é negativo, o resultado é velocidade escalar sem a necessidade do módulo.

Para o problema do monitoramento de camundongos, existem duas maneiras de calcular a velocidade, a média e a instantânea. Segundo Halliday [10]., a primeira pode ser útil considerando o comportamento geral de uma partícula durante certo intervalo, mas para descrever os detalhes do movimento, ela não é a mais indicada. Neste caso seria mais apropriado a obtenção de uma função matemática $v(t)$ que forneça a velocidade em cada ponto do movimento. Esta é denominada velocidade instantânea, muito mais precisa, porém, bem mais complicada de ser aplicada.

Neste trabalho será utilizada a velocidade média. Para diminuir os efeitos de seu uso, o cálculo da velocidade média será feito em curtos períodos de tempo, na casa de décimos de segundo. Desta maneira será possível observar todas as variações de movimento com uma perda controlada na qualidade dos dados. A partir deste ponto o termo velocidade será entendido como velocidade média, cuja fórmula é descrita abaixo.

$$\bar{v} = \frac{d}{(t_2 - t_1)} = \frac{d}{\Delta t} \quad (3.4)$$

Onde d é a distância euclidiana entre as posições x_2 e x_1 nos instantes t_2 e t_1 .

3.3.3 Aceleração

A velocidade de um camundongo pode variar durante um movimento, esta variação é conhecida como aceleração. Diferente da velocidade calculada na seção anterior, o cálculo da aceleração pode envolver índices negativos demonstrando que houve uma desaceleração, ou seja, a variação negativa revela que o animal está mais lento que na medição anterior.

A fórmula da aceleração é análoga ao da velocidade demonstrada na Equação 3.4. Ela é o quociente da variação da velocidade $\Delta v = v_2 - v_1$, pelo intervalo de tempo Δt .

$$\bar{a} = \frac{(v_2 - v_1)}{(t_2 - t_1)} = \frac{\Delta v}{\Delta t}. \quad (3.5)$$

3.4 Trabalhos Correlatos

Sistemas de rastreamento de camundongos e ratos utilizando técnicas de visão computacional são encontrados na literatura computacional. Em [4] por exemplo, é descrito um sistema que rastreia um roedor através de uma câmera. Ela captura as imagens para o processamento em um computador, permitindo assim a segmentação pelo uso da máscara, que é a imagem do fundo sem o objeto de interesse. O resultado desta cadeia de operações é uma imagem que preserva o formato do animal, possibilitando a extração de dados.

Branson, Rabaud e Belongie em [2] apresentam um sistema que trabalha com imagens extraídas lateralmente, a maioria posiciona câmeras acima do campo de observação para obter tomadas aéreas. Ele se concentra no rastreamento de múltiplos objetos, tratando de problemas como a identificação do objeto e do fundo da imagem e a sobreposição de objetos.

Em [38] o sistema de rastreamento preserva a câmera acima do campo, porém ela possui a propriedade de registrar imagens obtidas de raios infravermelhos, o que possibilitaria a gravação de imagens no escuro. Para o registro das imagens, o campo de observação deve ser iluminado por diodos emissores de raios infravermelhos, que não são visíveis nem para os humanos e nem para os ratos, preservando a rotina noturna do experimento.

O Ethovision [23] é sistema mais utilizado nos laboratórios. Este programa comercial possui além do rastreamento, um sistema de coletas automáticas de dados referentes a experiências como a velocidade, distância percorrida e o tempo de permanência em uma determinada região. O Ethovision já está sendo utilizado com frequência em experiências, e talvez no futuro, sistemas como ele sejam requisitos obrigatórios de uma pesquisa laboratorial.

Capítulo 4

Momentos Estatísticos

Momentos estatísticos ou momentos da imagem são utilizados na análise e interpretação de imagens. Neste capítulo, segue a descrição de seus principais tópicos e qual a sua utilidade neste trabalho.

4.1 Momentos

Os momentos da imagem são utilizados na visão computacional há muito tempo, especialmente na área de reconhecimento de padrões [3]. No processamento de imagens, muitas de suas tarefas são baseadas no reconhecimento de objetos, identificação de cenários e classificação de imagens. A maioria dos processos utiliza-se da forma e da intensidade dos pixels, seja eles coloridos ou em tons de cinza, para descrever um objeto, porém os métodos mais eficientes de descrição de imagens utilizam os momentos [37].

Liao [15] descreve em seu trabalho o teorema da unicidade de Hu. Historicamente Hu publicou o primeiro artigo que utilizou os momentos para a análise de imagens e representação de objetos. Sua abordagem foi baseada no trabalho dos matemáticos Boole, Cayley e Sylvester, todos do século XIX, sobre a teoria das formas algébricas. A teoria de Hu afirma que se $f(x,y)$ é um conjunto de valores contínuos não-nulos apenas na parte finita do plano (x,y) , então os momentos geométricos de todas as ordens existem para $f(x,y)$. Desta forma um grupo de momentos $\{m_{pq}\}$ é determinado unicamente por $f(x,y)$ assim como $\{m_{pq}\}$ determina de maneira única $f(x,y)$. Em outras palavras, de uma imagem $f(x,y)$ é possível extrair um único grupo $\{m_{pq}\}$ de momentos, que por sua vez só podem descrever a imagem $f(x,y)$.

Hu utilizou a combinação dos momentos geométricos para derivar um grupo de momentos invariantes, que permanece constante à translação, à rotação e ao redimensionamento, porém a reconstrução da imagem a partir

destes momentos é bastante difícil. Existem outros momentos como os momentos rotacionais, os momentos ortogonais e os momentos complexos que podem ser vistos no trabalho de Liao [15].

Na Figura 4.1(a) encontra-se um objeto em um plano cartesiano, ele representa a posição original e será representado em cinza nas demais gravuras. A translação pode ser observada em 4.1(b). A rotação é encontrada em 4.1(c) e por último a mudança nas dimensões em 4.1(d).

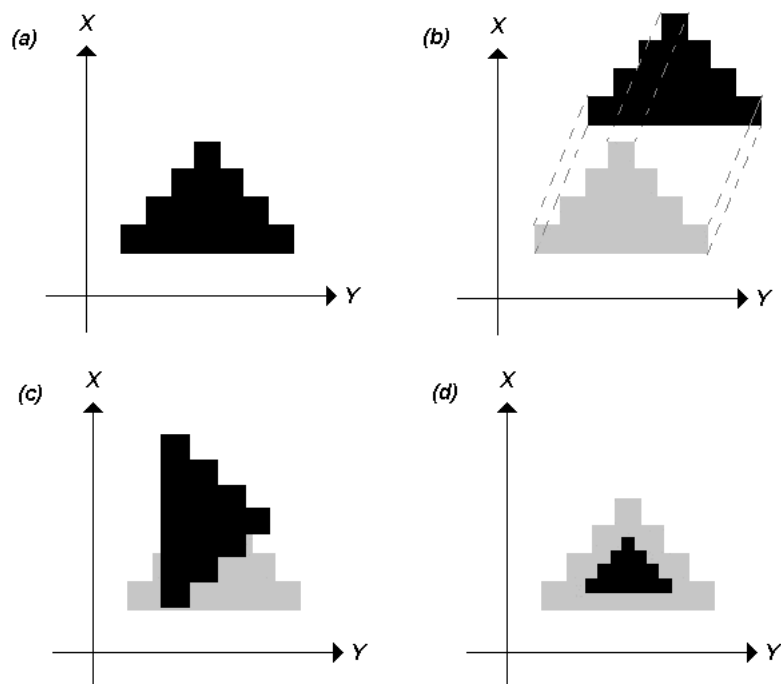


Figura 4.1: Operações as quais os momentos invariantes permanecem constantes à original

4.2 Momentos Geométricos

O momento geométrico de duas dimensões de ordem $(p + q)$ de uma função $f(x,y)$ é definida como:

$$M_{pq} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} x^p y^q f(x, y) dx dy \quad (4.1)$$

para $p, q = 0, 1, 2, \dots$

Um grupo de n momentos, é constituído de todos os M_{pq} onde $p + q \geq n$, ou seja, ele é composto por $\frac{1}{2} (n+1)(n+2)$ elementos. [15].

4.2.1 Momentos de ordem zero

Também chamados de Zeroth [29], o momento de ordem zero fornece a massa total de uma função ou imagem. No caso de uma imagem binária, $\{M_{00}\}$ representa a área total da imagem.

$$M_{00} = \sum_{x=a1}^{a2} \sum_{y=b1}^{b2} f(x, y) \quad (4.2)$$

Onde $a1, a2, b1, b2$ são os limites do plano de valores não-nulos ou os limites de uma imagem.

4.2.2 Momentos de primeira ordem

Os momentos de primeira ordem, $\{M_{10}\}$ e $\{M_{01}\}$, são as somas das coordenadas de x e y em uma figura binária. No momento $\{M_{10}\}$ a soma das coordenadas em x

$$M_{10} = \sum_{x=a1}^{a2} \sum_{y=b1}^{b2} x f(x, y) \quad (4.3)$$

No momento $\{M_{01}\}$ a soma das coordenadas em y

$$M_{01} = \sum_{x=a1}^{a2} \sum_{y=b1}^{b2} y f(x, y) \quad (4.4)$$

Estes momentos podem fornecer o centro de massa de uma figura.

Momentos centrais

O *centro de massa* é o ponto onde toda a massa de uma imagem pode ser concentrada sem mudar os momentos de primeira ordem. Suas coordenadas (\bar{x}, \bar{y}) podem ser encontradas da seguinte forma:

$$\bar{x} = \frac{M_{10}}{M_{00}} \quad \text{e} \quad \bar{y} = \frac{M_{01}}{M_{00}} \quad (4.5)$$

Ou seja, \bar{x} é a divisão da soma de todas as coordenadas em x pelo número de pixels, e \bar{y} a divisão da soma de todas as coordenadas em y pelo número de pixels, a grosso modo pode ser dito que o centro de massa possui a média das coordenadas de uma figura. Ele é muito utilizado como ponto de referência de um objeto.

4.2.3 Momentos de Segunda Ordem

Os momentos de segunda ordem $\{M_{02}, M_{11}, M_{20}\}$ fornecem a distribuição de um objeto [36]. Eles aproximam um objeto a uma elipse. O eixo maior e menor desta elipse são os auto-valores da matriz de co-variância, apresentada abaixo.

$$\begin{bmatrix} \mu_{20} & \mu_{11} \\ \mu_{11} & \mu_{02} \end{bmatrix}$$

O cálculo para μ_{pq} pode ser encontrado em 4.7.

O ângulo de rotação ou orientação entre o eixo x e o eixo mais próximo da elipse (maior ou menor) pode ser calculado como:

$$\alpha = \frac{1}{2} \arctan \left(\frac{2\mu_{11}}{\mu_{20} - \mu_{02}} \right) \quad (4.6)$$

α tem alcance $-45 \leq \alpha \leq 45$

4.3 Momentos centrais

Os momentos centrais de $f(x, y)$ são definidos como:

$$\mu_{pq} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} (x - \bar{x})^p (y - \bar{y})^q f(x, y) dx dy \quad (4.7)$$

No caso de uma imagem digital, a Equação 4.7 torna-se:

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q f(x, y) \quad (4.8)$$

4.4 Momentos invariáveis

Momentos invariáveis de uma imagem são aqueles que não variam devido a rotação, magnificação ou translação. Eles são baseados nas constantes algébricas [15]. Hu definiu 7 momentos invariantes baseados nos momentos de até 3ª ordem. Os 7 momentos invariáveis são:

$$\phi_1 = \mu_{20} + \mu_{02} \quad (4.9)$$

$$\phi_2 = (\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2 \quad (4.10)$$

$$\phi_3 = (\mu_{30} - 3\mu_{12})^2 + (3\mu_{21} - \mu_{03})^2 \quad (4.11)$$

$$\phi_4 = (\mu_{30} - \mu_{12})^2 + (\mu_{21} - \mu_{03})^2 \quad (4.12)$$

$$\begin{aligned} \phi_5 = & (\mu_{30} - 3\mu_{12})(\mu_{30} - \mu_{12})(\mu_{30} - \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2] \\ & + (3\mu_{21} - \mu_{03})(\mu_{21} + \mu_{03})[3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2] \end{aligned} \quad (4.13)$$

$$\begin{aligned} \phi_6 = & (\mu_{20} - \mu_{02})[(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2] \\ & + 4\mu_{11}(\mu_{30} + \mu_{12})(\mu_{21} + \mu_{03}) \end{aligned} \quad (4.14)$$

$$\begin{aligned} \phi_7 = & (3\mu_{21} - \mu_{03})(\mu_{30} + \mu_{12})[(\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2] \\ & - (\mu_{30} - 3\mu_{12})(\mu_{21} + \mu_{03})[3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2] \end{aligned} \quad (4.15)$$

Capítulo 5

Ferramentas

Neste capítulo segue a descrição de todas as ferramentas utilizadas na confecção dos módulos implementados.

5.1 ImageJ

O ImageJ, *Image Processing and Analysis in Java*, é um pacote para processamento de imagens escrito em JAVA. Inspirado no NIH Image para Macintosh, desenvolvido pela *Research Services Branch*(RSB)¹, este pacote possui código aberto e pode ser executado em qualquer computador que contenha uma maquina virtual Java, seja em forma de aplicativo ou *applet*.

Este pacote oferece recursos para exibição, edição, análise e processamento de imagens de 8, 16 e 32 bits em vários formatos como TIFF, GIF, JPEG, BMP, DICOM, FITS. Além disto, ele pode suportar pilhas, *stacks*, que são uma série de imagens compartilhando uma unica janela².

Outras vantagens do uso do ImageJ foram descrita em [27]. Entre elas, a vasta quantidade de algoritmos para: manipulação dos mais variados formatos de arquivo de imagens, detecção de bordas, melhoria de imagens, cálculos diversos (áreas, médias, centróides) e operações morfológicas.

O módulo implementado neste projeto utilizou classes do ImageJ, em especial o *ImagePlus* e *ImageProcessor*. Sem a utilização dos recursos fornecidos por este pacote, a implementação demoraria muito mais tempo ou pior, ela poderia não existir.

¹<http://rsb.info.nih.gov/nih-image/about.html>

²<http://rsb.info.nih.gov/ij/docs/intro.html>

5.2 JMF

O JMF é uma API desenvolvida pela Sun Microsystems, Intel e Silicon Graphics, onde a idéia principal foi fornecer recursos para que os aplicativos e *applets*, escritos sob a tecnologia JAVA pudessem suportar recursos de multimídia, áudio, vídeo e outras mídias baseadas em tempo. As mídias baseadas em tempo são aquelas que estão atreladas ao tempo e se modificam conforme ele transcorre, tome como exemplo os vídeos, músicas e animações³. Entre outras funções o JMF pode:

- Reproduz vários arquivos multimídia em applets ou aplicações JAVA. Os formatos suportados incluem *Microsoft Audio/Video Interleave (.avi)*, *Macromedia Flash 2 movies (.swf)*, *Future Splash (.spl)*, *MPEG Layer 3 Audio (.mp3)*, *Musical Instrument Digital Interface (MIDI; .mid)*, *vídeos MPEG-1 (.mpeg, .mpg)*, *QuickTime (.mov)*, *Sun Audio (.au)*, *áudio Wave (.wav)*, *AIFF (.aiff)* e *GSM (.gsm)*.
- Reproduz fluxos de dados de mídias vindos da Internet.
- Captura áudio e vídeo com seu microfone e câmera de vídeo, e armazena os dados em um formato suportado.
- Conversão dos formatos de mídia.
- Transmissão de áudio e vídeo em tempo real pela Internet

Através do JMF é possível criar aplicativos em Java que reproduzem, editam e capturam muitos tipos populares de mídia [7]. Neste projeto, o módulo implementado utilizou o JMF para retirar alguns quadros de vídeos do tipo *Microsoft Audio/Video Interleave (.avi)* e *QuickTime (.mov)*, para processamento.

5.3 Weka

O pacote Weka, *Waikato Environment for Knowledge Analysis*, é uma coleção de algoritmos de aprendizagem de máquina. Diversas pesquisas, que fazem uso do conhecimento extraído de dados, utilizam os recursos oferecidos por ele. Seus componentes facilitam a implementação de classificadores, no caso de aprendizagem supervisionada, e agrupadores, em não supervisionada. Além disso, é possível, com este pacote, obter facilmente resultados

³<http://java.sun.com/developer/technicalArticles/Media/mediaframework/>

estatísticos comparativos da execução simultânea de diversos programas de aprendizagem em domínios variados [18].

A interface gráfica do Weka permite o uso dos diversos algoritmos implementados, porém é possível que eles sejam invocados de um programa externo, escrito em JAVA. O Weka contém ferramentas para o pré-processamento de dados, classificação, regressão, agrupamento (*clustering*), regras de associação e visualização ⁴. Este pacote possui código aberto.

5.3.1 Arquivo Arff

O ARFF, *Attribute-Relation File Format*, é o formato de arquivo que o WEKA é capaz de ler e interpretar. O arquivo é do tipo texto, ASCII ⁵, e descreve a lista de instâncias e seus atributos. O módulo implementado gerará um arquivo neste formato para que o Weka possa ler e criar um classificador para os exemplos fornecidos.

Um arquivo Arff pode ser dividido em duas partes, o cabeçalho e os dados. O cabeçalho é composto do nome da relação, uma lista de atributos e seus tipos. Os tipos podem ser: numéricos, nominais (usados para definir as classes), string e data. Os dados representam os exemplos, cada exemplo ocupa uma linha e os valores de seus atributos são separados por vírgula ⁶. Um exemplo de um arquivo arff pode ser visto abaixo.

```
@RELATION exemploArff
@ATTRIBUTE minha_notas NUMERIC
@ATTRIBUTE classe {aprovado, reprovado}

@DATA
7.0, aprovado
2.5, reprovado
8.5, aprovado
```

⁴<http://www.cs.waikato.ac.nz/ml/weka>

⁵American Standard Code for Information Interchange

⁶<http://www.cs.waikato.ac.nz/ml/weka/arff.html>

Capítulo 6

Implementação das técnicas

Neste capítulo será descrito toda implementação das técnicas estudadas. Ele está dividido em três partes, a primeira delas demonstra a aquisição e o tratamento das imagens. A segunda descreve os métodos para a extração das características dinâmicas e, a terceira parte focaliza o problema da identificação postural, envolvendo métodos de aprendizagem automática.

6.1 Aquisição da imagens

A aquisição de imagens é o processo no qual é feita a coleta de imagens que alimentarão o sistema. A captura das imagens ocorreu nos laboratórios da UCDB sob supervisão do professor Albert Schiaveto de Souza do Centro de Ciências Biológicas e da Saúde, CCBS. Uma experiência com camundongos estava acontecendo em um platô. Uma webcam registrava todos os movimentos do camundongo, ela estava presa a um suporte a cerca de 1 metro de altura do platô, ficando perpendicular ao centro do campo da experiência conforme a Figura 6.1.

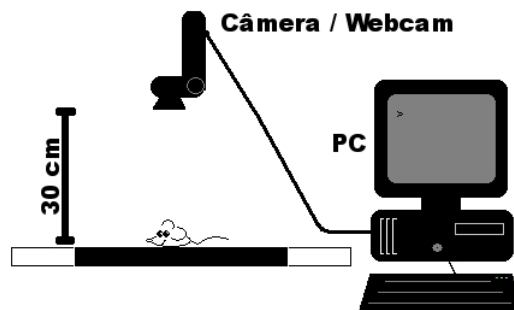


Figura 6.1: Captura de imagens

A imagens captadas pela webcam seguem o padrão da Figura 6.2, cujas dimensões são de 320 x 240 pixels de largura e altura consecutivamente.

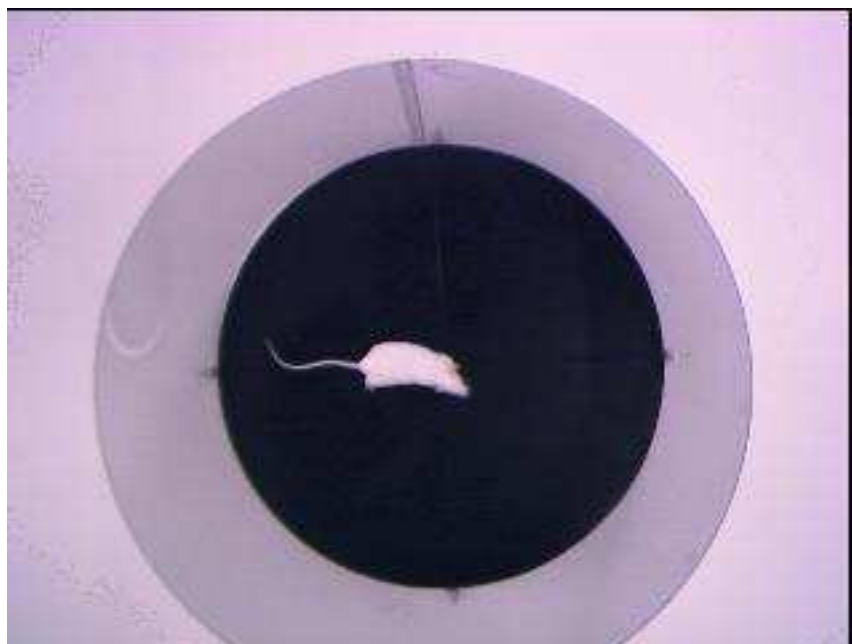


Figura 6.2: Imagem capturada pela Webcam

Um vídeo é formado de vários quadros que variam conforme o tempo, cada um deles pode se tornar uma imagem. No trabalho corrente, as imagens foram capturadas pelo JMF, pacote descrito na Seção 5.2, e modificadas pelas técnicas de visão computacional, que permitiram a extração das características. Este esquema foi descrito na Figura 6.3.

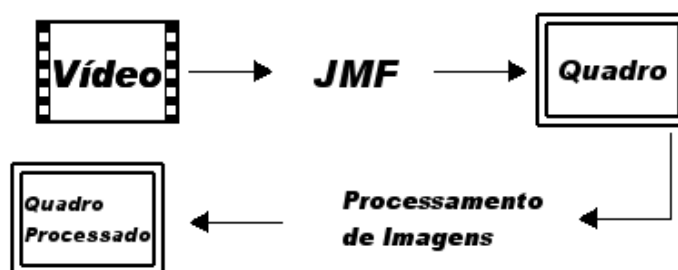


Figura 6.3: Esquema de captura de imagens

A primeira operação realizada na imagem é a conversão do padrão RGB para escala cinza, Figura 6.4(a). Em seguida, a nova imagem é submetida à limiarização, Figura 6.4(b). Este processo foi implementado seguindo as especificações da Seção 3.1.4

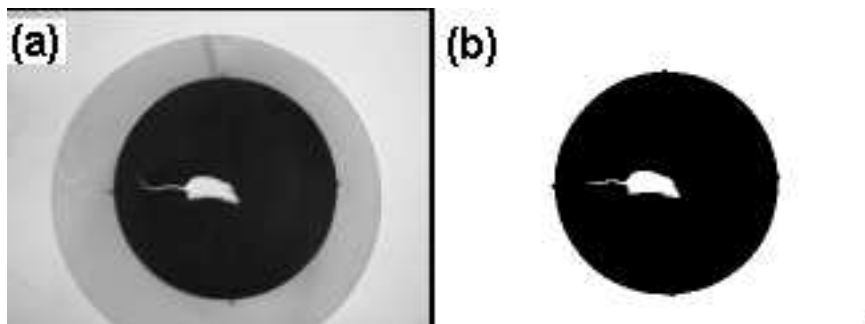


Figura 6.4: Operações na imagem

Para a segmentação do camundongo é necessária a criação de uma máscara para a subtração da imagem do fundo. Esta máscara é uma imagem do ambiente sem o camundongo como demonstrado na Figura 6.5(a). As mesmas operações da Figura 6.2 são aplicadas na máscara, o resultado pode ser encontrado na Figura 6.5(b).

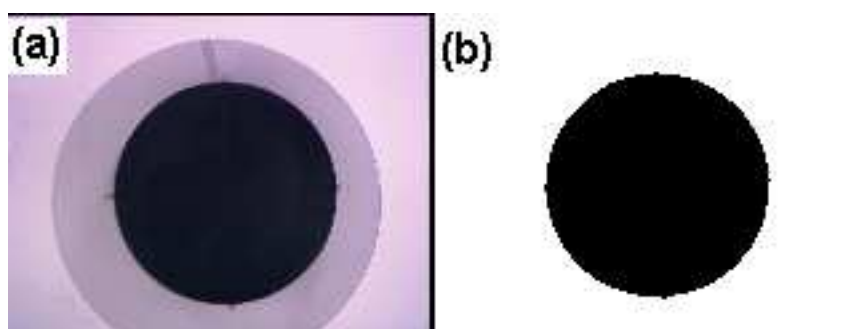


Figura 6.5: Operações na máscara

O resultado da subtração pode não ser perfeito deixando alguns pixels indesejáveis. Este problema pode ter vários motivos, um deles seria uma pequena mudança na posição da câmera ou uma diferença na quantidade de luz incidida no local.

A quantidade de pixels indesejáveis pode ser atenuada utilizando o processo de dilatação seguida de erosão Seção 3.1.5, pequenas falhas serão cobertas pela primeira operação Figura 6.6(b), porém ela provoca mudanças em toda a imagem. A segunda operação serve para diminuir estas modificações, que também não são desejáveis Figura 6.6(c).

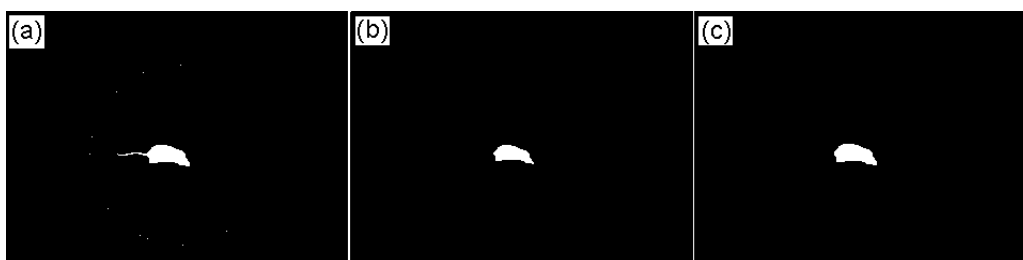


Figura 6.6: Dilatação e Erosão

A Figura 6.7 demonstra todo o processo de segmentação. A Figura 6.7 (a) é a imagem a ser tratada e a Figura 6.7 (b) a sua máscara. O resultado da subtração das duas é encontrada em 6.7(c) e após o processo de dilatação e erosão, a imagem final está pronta, Figura 6.7(d).

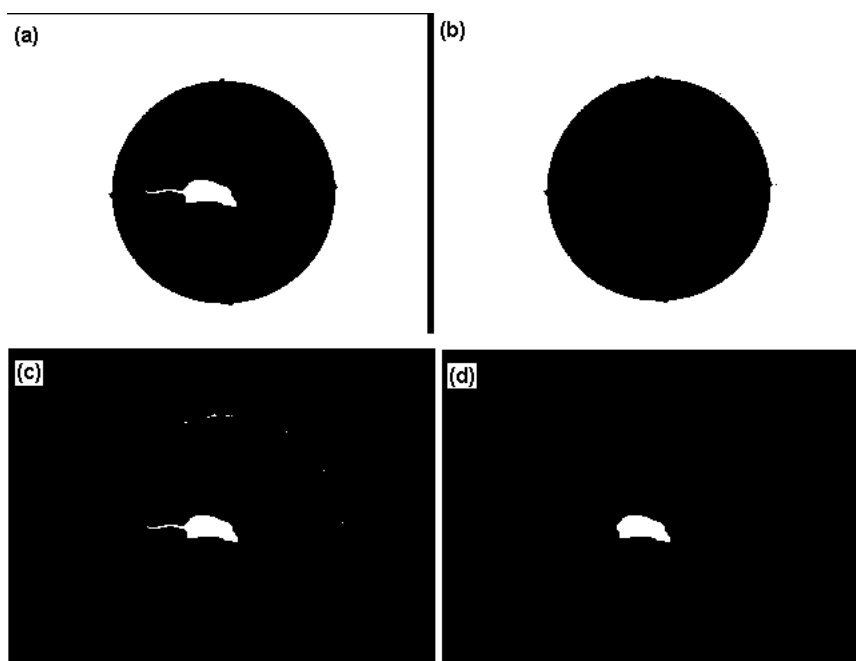


Figura 6.7: Subtração

6.2 Extração de características dinâmicas

Nesta seção está contida a descrição das características relacionadas ao movimento e posição espacial do camundongo. Elas estão divididas em:

- Distância Percorrida
- Velocidade
- Aceleração
- Tempo de permanência em uma região

Nas imagens segmentadas, conforme a seção anterior, o camundongo está sendo representado como um conjunto de pixels diferentes em relação ao meio. Para a obtenção dos dados é necessário o uso de um ponto de referência, uma unidade que represente a massa e que possua uma coordenada espacial, a esta é designada o nome de *centro de massa*.

Para a obtenção desta referência empregamos a fórmula encontrada na Seção (4.5), a representação gráfica é encontrada no ponto cinza da Figura 6.8.



Figura 6.8: Centro de massa de um camundongo

6.2.1 Distância Percorrida

O primeiro dado a ser coletado é a distância percorrida pelo camundongo durante um determinado período. O cálculo desta informação é realizado comparando a posição do centro de massa do camundongo em um instante t com a posição em $t-1$. A descrição das distâncias pode ser encontrada na seção 3.3.1, e a fórmula está transcrita abaixo considerando os pontos $p(x,y)$ e $q(s,t)$.

$$D_e(p, q) = \sqrt{(x - s)^2 + (y - t)^2} \quad (6.1)$$

Dois pontos são calculados, $p(2,4)$ e $q(4,6)$. Então,

$$D_e(p, q) = \sqrt{(2 - 4)^2 + (4 - 6)^2}$$

a distância entre p e q é de 2 pixels.

A medida em pixels pode não ser aceitável em experiências laboratoriais, a métrica deve corresponder a um comprimento de valores reais tais como metro ou centímetro. Na seção 3.1.2 foi visto que um pixel representa uma área estabelecida, para encontrar esta medida foi utilizado um modelo, cujas dimensões são conhecidas, na calibragem.

A Figura 6.9 ilustra uma calibragem, o escalímetro fornece medidas reais enquanto o traço amarelo conjunta os pixels necessários para a representação. Após uma regra de três é possível calcular a medida representada pelos conjuntos de pixels da figura. Levando-se em conta que a câmera possui uma posição fixa em relação ao ambiente da experiência, Apenas a calibragem inicial é necessária.



Figura 6.9: Calibragem

6.2.2 Velocidade

A velocidade é calculada com base na distância encontrada na seção anterior, as fórmulas são descritas na Seção 3.3.2. Os resultados são armazenados em uma tabela, onde mais tarde possam ser transpostos em um gráfico como na Figura 6.10.

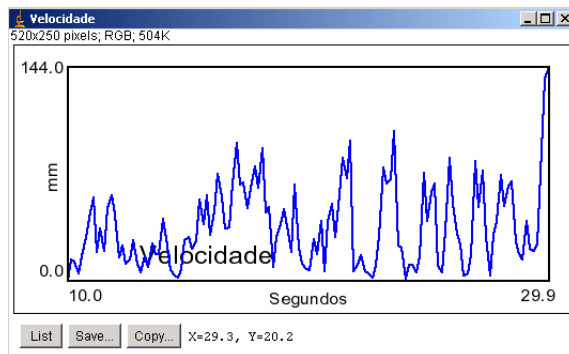


Figura 6.10: Gráfico da Velocidade

6.2.3 Aceleração

A aceleração depende do resultado do módulo da velocidade. É necessário um cuidado maior com a ordem dos cálculos, primeiro é preciso calcular a distância para depois a velocidade e então a aceleração. O primeiro valor necessita de mais tempo de processamento, ele possui operações com exponencial e raiz quadrada em sua fórmula. O resultado da aceleração também é inserido em uma tabela e transcrito em um gráfico, como demonstrado na Figura 6.11.

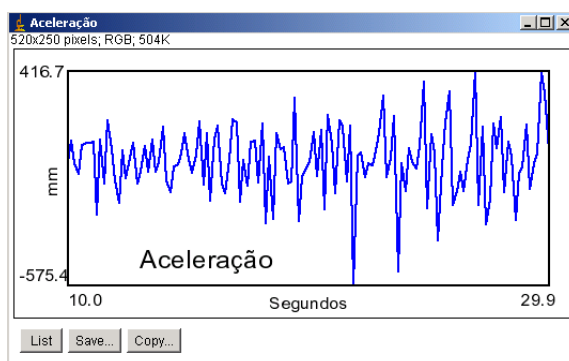


Figura 6.11: Gráfico da Aceleração

6.2.4 Tempo de permanência do camundongo em uma região

Nas experiências laboratoriais, às vezes, uma determinada região do ambiente se torna interessante. Comedouros e bebedouros, por exemplo, fornecem informações sobre os hábitos alimentares dos animais. A permanência dos roedores nas regiões mais escuras de uma gaiola podem sugerir medo.

Os dados obtidos pelo tempo de permanência em uma determinada região podem fornecer as mais diversas conclusões aos pesquisadores. Para isto é necessário a seleção prévia das áreas a serem observadas. A Figura 6.12 mostra a escolha da região à ser observada. Uma experiência com regiões marcadas pode ser visto na Figura 6.13.

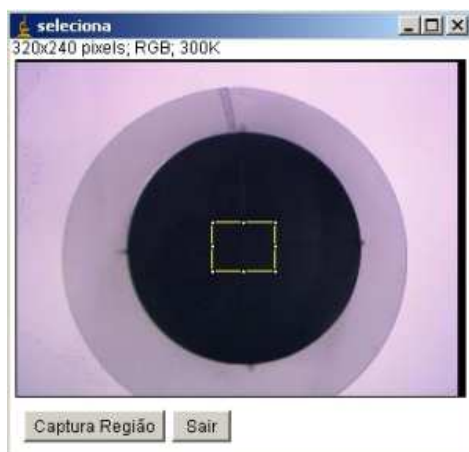


Figura 6.12: Seleção da Área de Observação

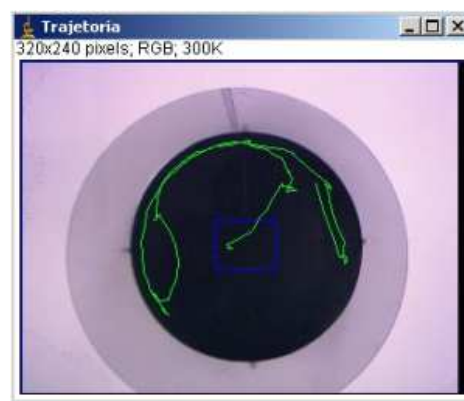


Figura 6.13: Experiência em Andamento

6.3 Extração de características posturais

A extração de características posturais está relacionada com as informações provenientes do formato de um objeto. Através delas é possível identificar algumas posições corporais que podem ajudar o pesquisador a determinar qual comportamento o camundongo está manifestando. Nas seções seguintes serão descritos os passos para a formação de um agente classificador, desde a captura das imagens de treinamento, passando pela formação do arquivo ARFF até a integração do classificador ao sistema.

6.3.1 Captura das Imagem de Treinamento e Armazenamento

O primeiro passo é decidir quais posições são relevantes para a experiência. Foram escolhidas as duas posições mais freqüentes encontrados nos vídeos, a normal (Figura 6.14) e a encolhido (Figura 6.15), para constituírem as classes deste trabalho, ou seja, qualquer imagem do camundongo deve ser interpretada ou como normal (classe 1) ou como encolhido (classe 2), dependendo da posição do animal.

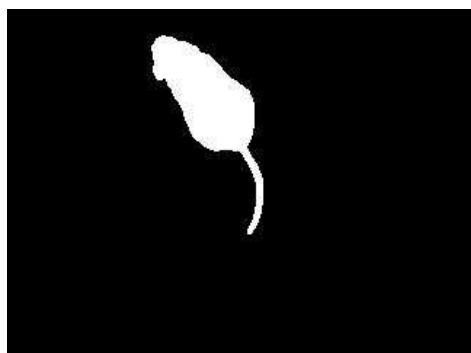


Figura 6.14: Normal

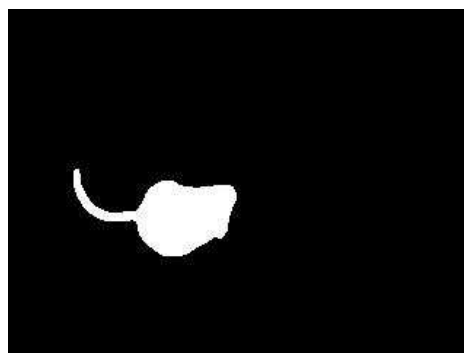


Figura 6.15: Encolhido

Após este processo, é necessária a captura de outras imagens do camundongo para o treinamento. Nesta fase deve-se tomar cuidado com dois aspectos, a generalização do sistema e a corretude dos exemplos de treinamento.

O sistema classifica as figuras conforme os exemplos que ele recebeu durante o treinamento. Quando uma nova figura é apresentada, o classificador verifica qual a classe que ela mais se aproxima. Um sistema treinado com imagens do camundongo em diversas posições e formas, dentro do conceito que caracteriza sua classe, poderia classificar corretamente uma nova imagem

mesmo se ela for diferente das que foram utilizadas no treinamento, ou seja, houve uma generalização.

A corretude dos exemplos é outro importante fator, a qualidade das imagens de treinamento é proporcional a precisão do agente classificador. Em uma visão superficial é possível dizer que se uma imagem de uma determinada classe for rotulada como outra, o classificador poderá classificar de forma errônea as figuras semelhantes a ela.

A tarefa de separar as imagens para o treinamento requer a supervisão de um ser humano conforme o esquema da Figura 6.16. Para facilitar este trabalho foi criado um *plugin*, denominado *rotulador*, que faz a captura e o tratamento¹ de um quadro em um vídeo, Figura 6.17.

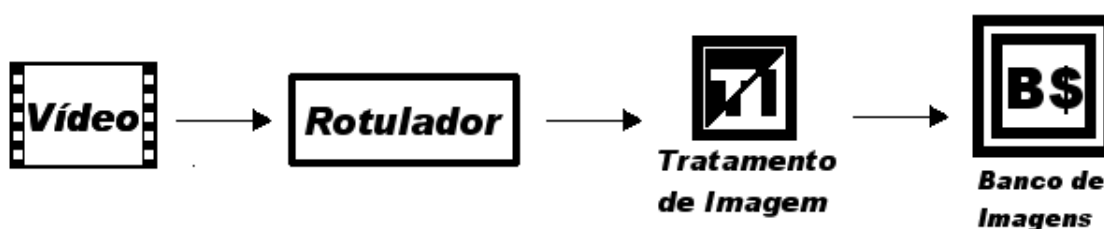


Figura 6.16: Esquema da formação do banco de imagens

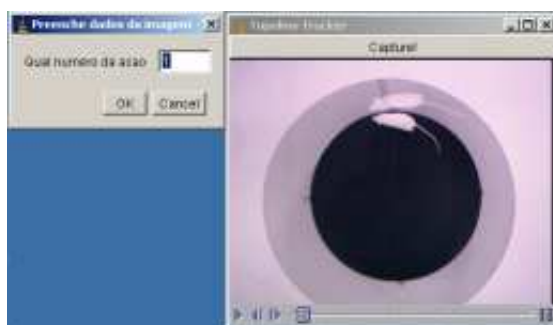


Figura 6.17: Rotulador

¹Binarização e segmentação

Outro problema que o *rotulador* resolveu está relacionado com a armazenagem das imagens no banco de imagens. As imagens devem conter três informações importantes, a sua identificação, a sua classe e o filme a qual ela foi extraída. A solução encontrada e implementada foi a utilização de 4 níveis de camadas de diretório, Figura 6.18.

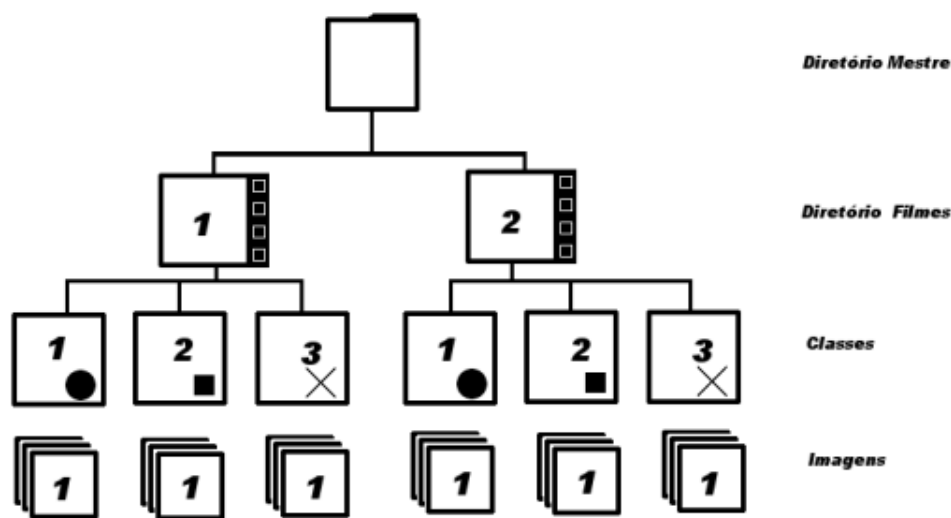


Figura 6.18: Estrutura do banco de imagens

Na primeira camada, a identificação das imagens está contida no próprio nome da figura, os nomes são dispostos na ordem crescente a partir do 1 e a seqüência não pode ser quebrada. Todas as imagens são agrupadas em diretórios que representam a classe a qual elas pertencem. Os diretórios “classe” formam a segunda camada, seus nomes também são numéricos e seguem uma ordem crescente sem quebra na seqüência.

A terceira camada é composta dos “diretórios filme”, eles representam a origem das imagens. Esta organização permite a inclusão de imagens de diversos filmes, mesmo que o banco de imagens já esteja estruturado. A quarta camada representa o próprio banco de imagem, aqui denominado “diretório mestre”.

6.3.2 Extração dos Momentos Invariantes e Geração do arquivo ARFF

Com a estruturação do banco de imagens, o próximo passo é a extração dos momentos invariantes. Eles serão utilizados como instâncias na criação de um arquivo do tipo arff, que será usado no processo de treinamento do classificador, Figura 6.19 .

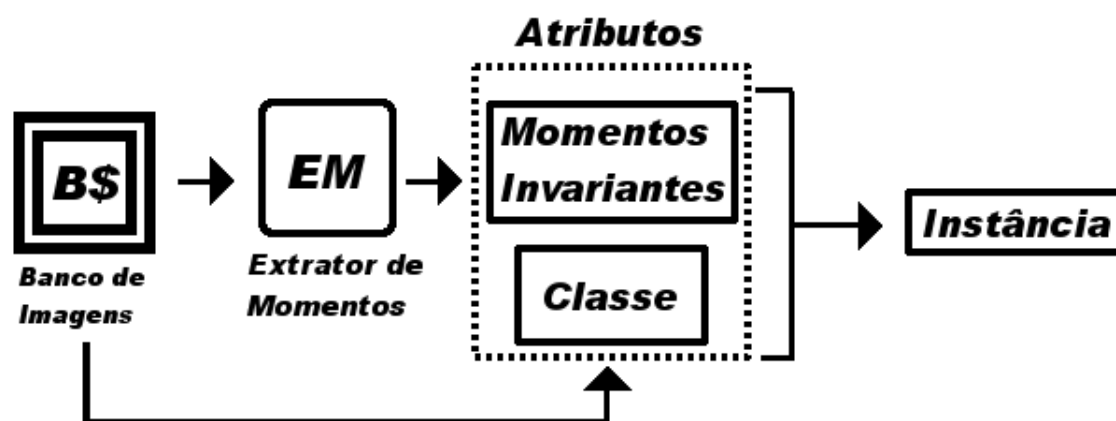


Figura 6.19: Criação de Instâncias

As imagens do banco serão lidas uma a uma pela função² *extrator de momentos*, que retornará o conjunto de momentos invariantes, definido na Seção 4.4.

Superficialmente pode ser dito que um conjunto de momentos só pode ter sido gerado por uma única imagem³ e, mesmo que ela tenha sofrido operações como a rotação, traslação e magnificação, seus valores não mudam.

²A função na verdade é uma classe JAVA, este termo foi substituído para evitar ambiguidade

³Explicação simplificada da Teoria da Unicidade de HU

Este fato pode ser comprovado observando a Figura 6.20, onde a imagem sofreu rotações de 90°, 180° e 270° graus.

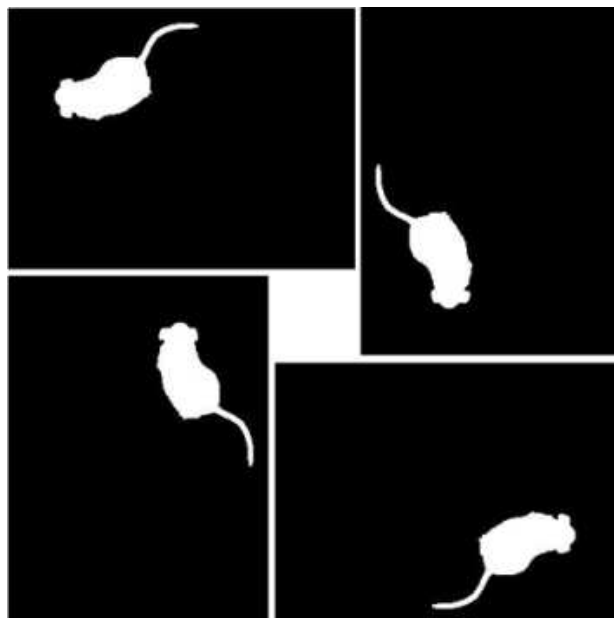


Figura 6.20: As 4 rotações de uma imagem

A Figura 6.21 demonstra a ação de uma translação, a imagem do rato foi deslocada se comparada as Figuras 6.20. Na Figura 6.22 contém a imagem do camundongo aumentada em 30%.

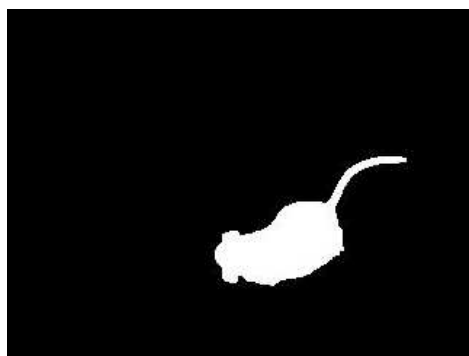


Figura 6.21: Deslocado

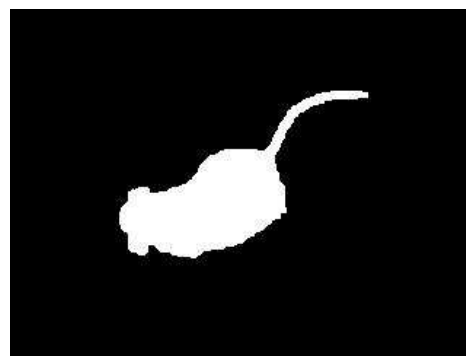


Figura 6.22: Aumentada em 30%

O resultado da coleta dos momentos invariantes das imagens pode ser visto na tabela abaixo. É possível observar que os valores não se diferem muito.

Momentos Invariantes							
Figura	I1	I2	I3	I4	I5	I6	I7
Normal	0.298	0.042	45.479	24.066	790.999	-0.605	91.069
Rotacionada em 90°	0.298	0.042	44.638	23.304	746.298	-0.431	89.676
Rotacionada em 180°	0.298	0.042	44.828	23.531	759.059	-0.606	89.155
Rotacionada em 270°	0.298	0.042	44.548	23.323	746.496	-0.428	89.116
Deslocada	0.303	0.043	47.256	24.990	853.295	-0.651	97.278
Aumentada em 30%	0.283	0.036	60.923	32.124	1411.755	-0.718	163.112

Os momentos invariantes serão usados juntamente com as classes na formação das instâncias. As classes serão obtidas diretamente do banco de imagens, através do sistema de camadas de diretório. Com as instâncias prontas, é possível a criação do arquivo ARFF através do *gerador de arff*, módulo desenvolvido neste projeto, Figura 6.23. A descrição deste arquivo está localizado na seção 5.3.1 e será utilizado no treinamento do classificador.

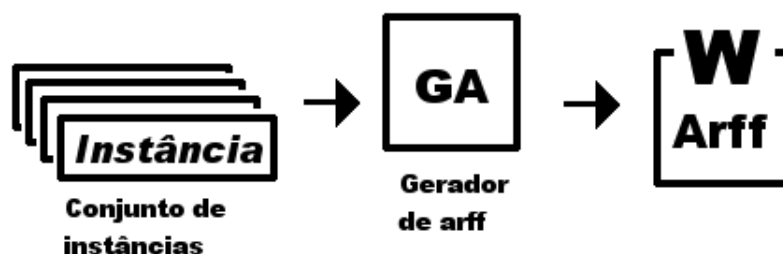


Figura 6.23: Formação do Arff

6.3.3 O classificador

Com a criação do arquivo ARFF de treinamento, o sistema está pronto para gerar um classificador. O pacote WEKA, descrito na Seção 5.3, possui uma grande quantidade de algoritmos de aprendizagem implementados. O trabalho em questão utilizará um deles para classificar as imagens entre as classes “Normal” e “Encolhido”. A Figura 6.24 demonstra a fase de treinamento.

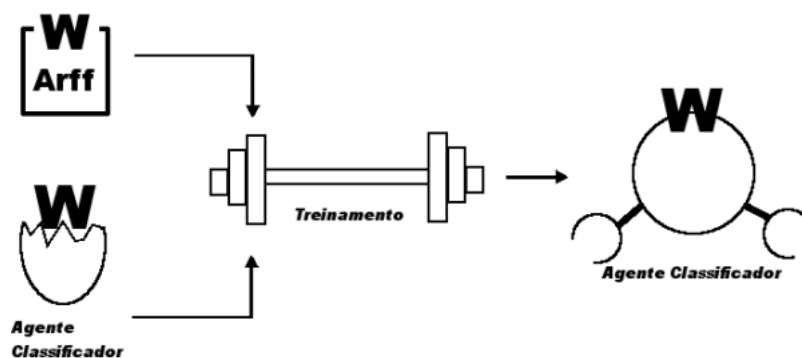


Figura 6.24: Treinamento

Depois deste processo, o agente classificador está preparado para atender às necessidades deste projeto. Quando uma instância entra no sistema, o classificador decidirá qual a classe da imagem que gerou aquela instância. A Figura 6.25 ilustra esta ação.

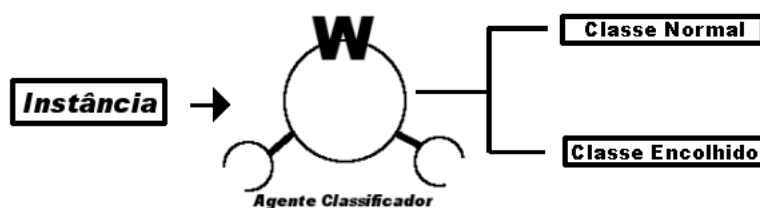


Figura 6.25: Classificador Atuando

Capítulo 7

Experimentos, Resultados e Análise.

Neste capítulo está contida a descrição dos experimentos utilizando o módulo, os resultados apresentados por ele e a análise dos dados.

7.1 Experimentos

Os testes com o módulo sucederam em um platô de 20 x 20 cm. Eles foram realizados em modo *off-line*, ou seja, as imagens não foram processadas em tempo real e o vídeo foi gerado. A escolha por este modo de trabalho facilitou a coleta dos quadros para o Banco de Imagens, pois as ações do camundongo não estariam mais atreladas ao tempo real.

O conjunto de treinamento é composto por 101 imagens sendo destas, 43 da classe “Normal” e 58 da classe “Encolhido”. A câmera utilizada foi a PowerShot A-75 da fabricante Canon, instalada a 30 cm acima do platô. Foram realizadas 4 tomadas de 1:30 minutos cada.

O animal utilizado para teste foi obtido do biotério da UCDB. O camundongo possui pêlos claros, demonstra saúde perfeita e foi nomeado de “Hoax”. A Figura 7.1 ilustra como foi realizado o experimento.

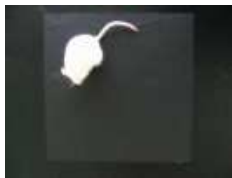


Figura 7.1: Experimento

7.2 Resultados

O módulo foi capaz de extrair as características dinâmicas com sucesso. A Figura 7.2 demonstra como o módulo exibe o resultado da extração.

Tabela de Estatísticas								
File	Edit							
	Tempo	x1	y1	x2	y2	distancia	velocidade	aceleração
29	14.251	193.261	80.257	193.766	81.346	1.200	7.904	-138.886
30	14.403	193.111	79.582	193.261	80.257	0.692	4.556	-22.051
31	14.555	192.992	79.167	193.111	79.582	0.432	2.843	-11.284
32	14.707	192.533	80.351	192.992	79.167	1.270	8.366	36.378

Figura 7.2: Características Dinâmicas

As informações sobre a permanência do camundongo nas regiões estão dispostas na Tabela de Regiões. A Figura 7.3 demonstra como é armazenada esta informação. Cada região é representada por uma coluna, o número 1(um) indica que o camundongo está na região e o número 0(zero) que não está.

File	Edit				
	Label	0	1	2	3
5	10.61	1	1	0	0
6	10.78	1	1	0	0
7	10.91	1	1	0	0
8	11.06	1	1	0	0
9	11.21	1	1	0	0
10	11.37	1	1	0	0
11	11.52	1	1	0	0
12	11.67	1	0	0	0
13	11.82	1	0	0	0
14	11.97	1	0	0	0

Figura 7.3: Tabela de Regiões

Na extração das características posturais a precisão do agente classificador deve ser avaliada. O pacote Weka contém uma ampla gama de classificadores, 59 dos algoritmos foram testados empregando a validação cruzada ¹. Este tipo de validação estima a precisão de um classificador, o seu uso é recomendado quando o grupo de dados é pequeno ². Ele separa as instâncias em vários pares de grupos de treinamento e teste. Cada um dos pares recebe

¹ *Cross-Validation*

² <http://hartford.lti.cs.cmu.edu/classes/95-779/Lectures/09-WEKAB.pdf>

um novo classificador, a média de acerto deles será a taxa de precisão do classificador como um todo. A Figura 7.4 ilustra o esquema de uma Validação Cruzada.

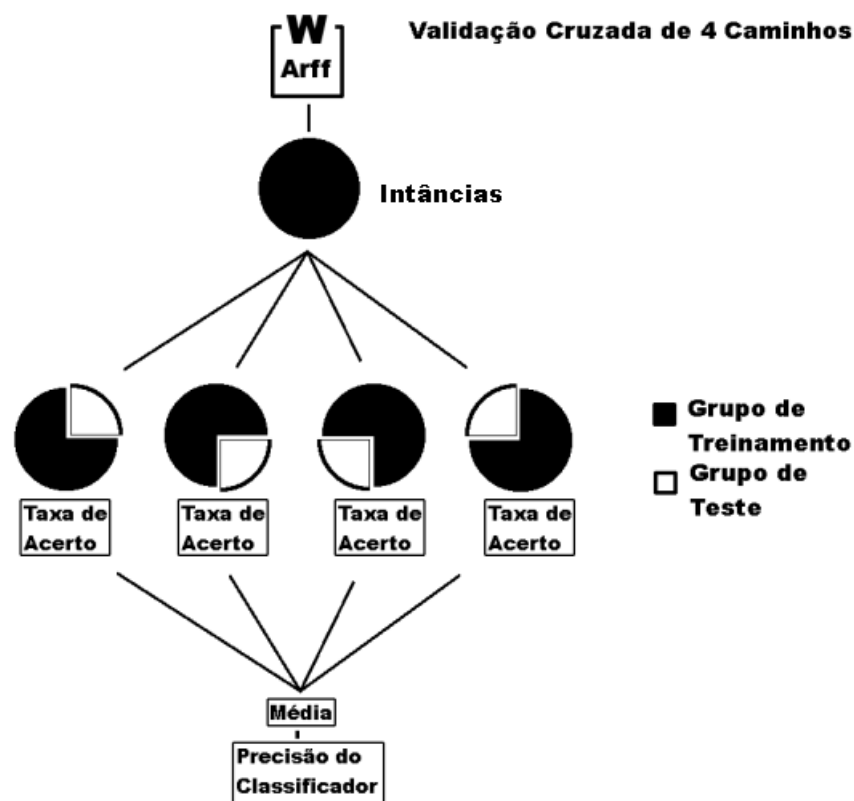


Figura 7.4: Validação Cruzada

Os 7 classificadores mais encontrados na literatura do gênero foram escolhidos para serem testados. O objetivo destes testes é a escolha do classificador que melhor adapte ao contexto do sistema. Os critérios escolhidos para seleção são:

- Porcentagem de Acerto: Porcentagem de classificações corretas
- Falso Positivo: A hipótese de um elemento ser rotulado como uma classe mas de fato pertence a outra
- Tempo de Treinamento: Tempo do treinamento do classificador
- Tempo de Teste: Tempo que um classificador treinado leva para classificar uma instância

A tabela abaixo contém os nomes e abreviatura, utilizadas neste projeto, dos classificadores selecionados.

Classificadores	
Sigla	Classificador
BNet	BayesNet
NBayes	NaiveBayes
MPcp	MultilayerPerceptron
KStar	KStar
J48	J48
DTable	DecisionTable
SLog	SimpleLogistic

Os teste foram realizados no WEKA e os resultados estão dispostos nas tabelas a seguir.

Data Set	(BNet)	(NBayes)	(MPcp)	(KStar)	(J48)	(DTable)	(SLog)
trainingSet	97.55	98.02	98.73	98.92	98.81	98.91	99.01

◦, • statistically significant improvement or degradation

Tabela 7.1: Porcentagem de acerto

Data Set	(BNet)	(NBayes)	(MPcp)	(KStar)	(J48)	(DTable)	(SLog)
trainingSet	0.03	0.04	0.00	0.00	0.00	0.00	0.00

◦, • statistically significant improvement or degradation

Tabela 7.2: Falso Positivo

Data Set	(BNet)	(NBayes)	(MPcp)	(KStar)	(J48)	(DTable)	(SLog)
trainingSet	0.00	0.00	0.46	◦	0.00	0.00	0.01

◦, • statistically significant improvement or degradation

Tabela 7.3: Tempo de Treinamento

Data Set	(BNet)	(NBayes)	(MPcp)	(KStar)	(J48)	(DTable)	(SLog)
trainingSet	0.00	0.00	0.00	0.04	◦	0.00	0.00

◦, • statistically significant improvement or degradation

Tabela 7.4: Tempo de Teste

7.3 Análise

Os dados obtidos pelo extrator de característica dinâmica não puderam ser avaliados, pois não foi encontrado um método prático e confiável para aferir os resultados com as métricas reais. Em outras palavras, não foi obtido um mecanismo, como um radar, por exemplo, para obtenção de dados que pudessem validar os valores calculados pelo módulo.

Por outro lado, as informações fornecidas pelo WEKA podem esclarecer algumas características do classificador. Na Tabela 7.1 que carrega os dados da porcentagem de acerto, é possível verificar a estreita margem da diferença entre os algoritmos. Todos eles atingem taxa superiores a 97% sendo declarados tecnicamente empatados.

A Tabela 7.2 demonstra as chances individuais que cada exemplo tem de sofrer uma classificação errônea. Esta informação é importante, pois muda o foco da atenção. A visão macro de um classificador é trocada por uma micro, do ponto de vista de um exemplo. Observa-se que o classificador *NaiveBayes* possui uma porcentagem de acerto maior que o *BayesNet*, porém um exemplo dele possui 1% de chances a mais de ser classificado de forma errada em relação ao outro classificador.

O tempo de treinamento pode ser encontrado na Tabela 7.3. Os resultados mostram que o MultilayerPerceptron é o algoritmo que consome mais tempo de treinamento. O tempo de teste, disposto na Tabela 7.4 é designado ao período de tempo que um classificador treinado usa na classificação de uma

entrada. O algoritmo *k-star* apresentou maior índice.

A análise final é realizada comparando as informações das tabelas, é possível concluir que nas condições fornecidas por este projeto, os classificadores possuem desempenhos parecidos. As duas primeiras tabelas indicam um empate estatístico entre eles, porém nas tabelas de tempo o Multilayer-Perceptron e o *k-star* demonstraram um desempenho inferior em relação aos demais. Entre os 5 restantes, o J48 foi escolhido como classificador por ser o mais conhecido, ele é a implementação do C4.5, cuja descrição pode ser encontrada em [21].

Capítulo 8

Considerações finais

Sistemas computadorizados de análise de comportamento de animais não são novidades para o mundo. Os ramos da pesquisa que possibilitaram a criação destes sistemas têm origens anteriores ao advento dos computadores, já nesta época era visível os benefícios desta ferramenta, mesmo quando o preço dos componentes eletrônicos eram superiores às horas pagas aos pesquisadores, que realizavam a coleta de dados de forma manual.

Estas ferramentas se firmaram como tendência quando surgiram os modelos que utilizavam vídeos no auxílio à coleta de dados. Desta forma era possível controlar o tempo da experiência no apertar de um botão. Nos dias correntes, o poder computacional permite não apenas que os vídeos sejam assistidos, mas também que informações sejam extraídas deles sem a interferência humana.

O presente trabalho foi baseado nos sistemas de rastreamento por vídeo. Primeiramente foi feito um levantamento bibliográfico buscando as técnicas agregadas à estas ferramentas. Foram consultados livros, artigos e textos na Internet relacionados com estes sistemas.

A maioria destes sistemas oferece apenas os dados relacionados com os movimentos dos animais, tais como distância percorrida e tempo de permanência em uma determinada região, porém isto serviu de inspiração para a primeira parte do projeto, a implementação do extrator de características dinâmicas.

A segunda parte deste projeto envolveu a classificação postural do animal. Este segmento do trabalho exigiu o estudo de uma forma de representar o formato do camundongo, este problema foi resolvido com a utilização dos momentos da imagem. O próximo passo foi a escolha de um classificador que pudesse classificar a posição do animal através do conjunto de momentos extraídos de uma imagem.

Após a conclusão da implementação do módulo de extração de caracte-

terísticas posturais, foi possível notar as limitações deste sistema. Para que a classificação postural obtivesse sucesso, a câmera que realiza a captura do vídeo teve que ser instalada a poucos centímetros da área da experiência.

Todo o material gerado e os módulos implementados serão disponibilizados para o grupo de desenvolvimento do projeto Topolino, cumprindo assim um dos objetivos deste trabalho.

Trabalhos futuros devem aperfeiçoar a segmentação e o uso dos momentos da imagem. Desta forma será possível o uso de imagens em tons de cinza e coloridas, que fornecerão informações sobre a cor, textura e intensidade do objeto.

Referências Bibliográficas

- [1] Serge Belongie, Chad Carson, Hayit Greenspan, e Jitendra Malik. Color- and texture-based image segmentation using EM and its application to content-based image retrieval. In *Proceedings of the Sixth International Conference on Computer Vision*. 1998.
- [2] Kristin Branson, Vicent Rabaud, e Serge Belongie. Three brown mice: See how they run. *IEEE International Workshop on VS-PETS*, 2003.
- [3] François Chaumette. Image moments: A general and useful set of features for visual servoing.
- [4] C.J.Twining, C.J.Taylor, e P.Courtney. *Robust tracking and posture description for laboratory rodents using active shape models*, 2001.
- [5] D. Comaniciu e P. Meer. Robust analysis of feature spaces: Color image segmentation, 1997.
- [6] John W. Dalya e Bertil B. Fredholm. Caffeine - an atypical drug of dependence. *Elsevier Science Ireland Ltd.*, 1998.
- [7] H. M. Deitel e P. J. Deitel. *Java Como Programar*. Bookman, 2003.
- [8] Roger Sedlacek Fibiger. Estudo comparativo de técnicas de visão computacional para a detecção de pele humana, 2004.
- [9] Rafael C. Gonzalez e Richard E. Woods. *Processamento de imagens digitais*. Edgard Blücher, 2000.
- [10] David Halliday, Robert Resnick, e Kenneth S. Krane. *Física 1*. LTC, 1996.
- [11] K Hattori, H Lee, PD Hurn, BJ Crain, RJ Traystman, e AC DeVries. Cognitive deficits after focal cerebral ischemia in mice. *American Heart Association, Inc*, 2000.

- [12] Henk J. A. M. Heijmans. Mathematical morphology: A modern approach in image processing based on algebra and geometry. *SIAM Review*, 37(1):1–36, 1995.
- [13] Boulukos Kim E e Philippe Pognonec. Mice a program to track and monitor animals in animal facilities. *BMC Genetics*, 2001.
- [14] Andrew Laine e Chun-Ming Chang. De-noising via wavelet transforms using steerable filters. In *ISCAS*, páginas 1956–1959. 1995.
- [15] Simon Xinmeng Liao. Image analysis by moments, 1993.
- [16] D. Lipkind, A. Sakov, N. Kafkafi, G.I. Elmer, Y. Benjamini, e I. Golani. New replicable anxiety-related measures of wall vs center behavior of mice in the open field. *J Appl Physiol*, 2004.
- [17] B. K. Lucas, C. J. Ormandy, N. Binart, R. S. Bridges, e P. A. Kelly. Null mutation of the prolactin receptor gene produces a defect in maternal behavior. *Endocrinology*, 1998.
- [18] Álvaro Roberto Silvestre Fialho. Estudo de técnicas de rastreamento das mãos para o desenvolvimento de interfaces homem-máquina, 2004.
- [19] S.K. Mani, J.M. Allen, J.P. Lydon, B. Mulac-Jericevic, J.D. Blaustein, F.J. DeMayo, O. Conneely, e B.W. O'Malley. Dopamine requires the unoccupied progesterone receptor to induce sexual behavior in mice. *Endocrine Society*, 1996.
- [20] Aline Miltersteiner, Diego Miltersteiner, Nelson Pereira Filho, Aline Rolim Frota, Pedro Bins Ely, Cláudio Galeano Zettler, e Norma P. Marroni. Uso de quercetina a longo prazo em ratos cirróticos. *Acta Cirúrgica Brasileira*, 2003.
- [21] Tom M. Mitchell. *Machine learning*. Nova York : McGraw-Hill, 1997.
- [22] Peter Montague. Pesticides and aggression. Relatório técnico, Environmental Research Foundation, 1999.
- [23] Lucas P.J.J Noldus, Andrew J Spink, e Ruud A J Tegelenbosch. *Ethovision: A Versatile video tracking system for automation of behavior experiments*, 2001.
- [24] Petra Perner. Motion tracking of animals for behavior analysis. Relatório técnico, Institute of Computer Vision and Applied Computer Sciences, 2001.

- [25] Giselle M. Petzinger e Michael W. Jakowec. Animal models of basal ganglia injury and degeneration and their application to parkinson's disease research. Relatório técnico, Department of Neurology, Keck School of Medicine, University of Southern California, 2004.
- [26] Adriano Donizete Pila. Seleção de atributos relevantes para aprendizado de máquina utilizando a abordagem de rough sets, 2001.
- [27] Hemerson Pistori e Mauro Conti Pereira. Integração dos ambientes livres weka e imagej na construção de interfaces guiadas por sinais visuais. *V Workshop sobre Software Livre*, 2004.
- [28] T. Pun, J. Bost, R. Milanese, C. Rauber, e S. Startchik. Selecting relevant information and delaying irrelevant data for objects recognition, 1994.
- [29] Lourena Rocha, Luiz Velho, e Paulo Cezar Carvalho. Image moments-based structuring and tracking of objects. *In Proceedings of SIBGRAPI 2002 - XV Brazilian Symposium on Computer Graphics and Image Processing. SBC - Sociedade Brasileira de Computacao*, 2002.
- [30] Alena V. Savonenko, Guilian M. Xu, Donald L. Price, David R. Borchelt, e Alicja L. Markowska. Normal cognitive behavior in two distinct congenic lines of transgenic mice hyperexpressing mutant appsw. *Elsevier Science*, 2003.
- [31] M. Souto, A. Lorena, A. Delbem, e A. de Carvalho. Técnicas de aprendizado de máquina para problemas de biologia molecular. *III Jornada de Mini-Curso de Inteligência Artificial*, 2001.
- [32] Nobuo Sugo, Patricia D. Hurn, M. Brigid Morahan, Kimihiko Hattori, Richard J. Traystman, e A. Courtney DeVries. Social stress exacerbates focal cerebral ischemia in mice. *American Heart Association, Inc*, 2002.
- [33] M. Tuceryan. Moment-based texture segmentation. *Elsevier Science Inc*, 15(7):659–668, July 1994.
- [34] Jeanne M. Wehner. Mouse models of psychiatric disorders. Relatório técnico, JAX - The Jackson Laboratory, 2003.
- [35] S.R. Wersinger, K. Sannen, C. Villalba, D.B. Lubahn, E.F. Rissman, e G.J.De Vries. Masculine sexual behavior is disrupted in male and female mice lacking a functional estrogen receptor small alpha, greek gene. *PubMed*, 1997.

-
- [36] H. H. Yakali, L. Dorst, e B. J. A. Kröse. Pose characterization by independent moment-based image features of planar objects. Relatório técnico, Kruislaan 403, 1098 SJ, Amsterdam, The Netherlands, 1997.
- [37] Feng Zhou e Peter Kornerup. Computing moments by prefix sums. Relatório Técnico PP-1995-31, 1, 1995.
- [38] J. Brooks Zurn, Xianhua Jiang, e Yuichi Motai. Video-based rodent activity measurement using near-infrared illumination. *IMTC - Instrumental and Measurement Technology Conference*, 2005.
- [39] Jane Brooks Zurn, Drew Hohmann, Steven I. Dworkin, e Yuichi Motai. *A real-time rodent tracking system for both light and dark cycle behavior analysis*, 2001.