

# **Extração de características do caminho percorrido por um rato na tarefa mnemônica do labirinto aquático de Morris**

**Moacir Guedes Oliveira<sup>1</sup>, Mauro Miazaki<sup>1\*</sup>, Jotair Elio Kwiatkowski Junior<sup>1†</sup>**

<sup>1</sup>Departamento de Ciência da Computação – DECOMP  
Universidade Estadual do Centro-Oeste - UNICENTRO  
R. Simeão Varela de Sá, 03 - Vila Carli, Guarapuava - PR, 85040-080

guedesmoacir12@gmail.com, maurom@unicentro.br, jotairkw@yahoo.com.br

**Abstract.** In the Laboratory of Neuropsychopharmacology, in the Department of Pharmacy at UNICENTRO, studies are carried out about memory and spatial learning in rats using the Morris Water Maze (MWM). Currently, the path traced by a rat in the MWM is analyzed in a manual way. In order to automate such procedure, the rat tracking was implemented in a Scientific Initiation project. In addition to that project, in this work, the objective was to implement feature extraction techniques in order to obtain quantitative data from the recognized path: distance, time, average speed, swimming path drawing and occupancy density. The development was done in the Visual Studio 2017 programming environment using the C ++ programming language and the OpenCV, an open source library.

**Resumo.** No Laboratório de Neuropsicofarmacologia, no Departamento de Farmácia da UNICENTRO, são realizados estudos sobre memória e aprendizagem espacial em ratos utilizando o Labirinto Aquático de Morris (LAM). Atualmente, o caminho percorrido por um rato no LAM é analisado de forma manual. Visando a automatização desse procedimento, o rastreamento de rato foi implementado em um projeto de Iniciação Científica. Em complemento a esse projeto, neste trabalho, o objetivo foi implementar técnicas de extração de características para obter dados quantitativos a partir do caminho reconhecido: distância percorrida, tempo de percurso, velocidade média, desenho da trajetória de nado e densidade de ocupação. O desenvolvimento foi realizado no ambiente de programação Visual Studio 2017 utilizando a linguagem de programação C++ e a biblioteca open source OpenCV.

## **1. Introdução**

O Labirinto Aquático de Morris (LAM) é uma ferramenta amplamente aceita para avaliar a capacidade de aprendizagem e memória espacial em pequenos roedores, sendo provavelmente o paradigma de aprendizagem mais utilizado nessa tarefa [Bures et al. 1997, Redish e Touretzky 1998, De Hoz et al. 2004, Schimanski e Nguyen 2004]. Essa ferramenta pode ser utilizada em estudos sobre a influência de medicamentos no desempenho neurocognitivo e sobre transtornos, como a Doença de Alzheimer. No Laboratório de

\*Orientador

†Coorientador

Neuropsicofarmacologia, do Departamento de Farmácia (DEFAR), da Universidade Estadual do Centro-Oeste (UNICENTRO), há uma sala com um LAM, onde são realizados estudos comportamentais usando ratos como sujeitos experimentais. Atualmente, o comportamento dos ratos é avaliado de forma manual pelos pesquisadores, o que torna o processo demorado, cansativo, impreciso, subjetivo e limitado quanto às possibilidades de extração de dados para análise. Portanto, a automatização dos procedimentos manuais possibilitará auxiliar na resolução desses problemas.

O LAM consiste de um tanque preenchido com água e uma plataforma escondida logo abaixo da superfície da água. A tarefa nesse tanque é o treinamento de ratos (ou outros pequenos roedores) para escapar da água [De Hoz et al. 2004, Schimanski e Nguyen 2004]. Essa tarefa é realizada da seguinte forma: um rato é inserido no tanque de água; inicialmente, a localização da plataforma é desconhecida pelo rato, fazendo com que ele precise nadar para encontrá-la; nas próximas tentativas, espera-se que o rato seja capaz de se lembrar aproximadamente onde está a plataforma. A localização dessa plataforma só pode ser encontrada utilizando-se de pistas espaciais externas ao tanque [Blokland et al. 2004]. Análises comportamentais do aprendizado na tarefa do LAM indicam que há a mobilização de um grupo de processos cognitivos envolvidos por um lado com o armazenamento e evocação das informações visuais. E, por outro lado, com o planejamento e estratégias de navegação [Morris 1989].

Durante o aprendizado espacial, o parâmetro quantitativo mais simples que evidencia a gradual mudança do desempenho do animal em encontrar o local da plataforma é o declínio no tempo que esse animal leva para encontrar esse local. Esse tempo é denominado latência de escape [Morris 1984]. Entretanto, como pode haver muitos outros fatores que contribuem para uma dada latência de escape, além da taxa de aprendizagem espacial, torna-se necessário conduzir outras análises de desempenho para se obter um cenário mais detalhado. Dentre essas análises, o teste de transferência é um dos mais importantes e utilizados, pois fornece um parâmetro, independente da latência de escape e da velocidade de nado, da tendência de o animal persistir na procura pelo local onde a plataforma se encontrava durante todo o treino [Stewart e Morris 1993, Spink et al. 2001]. O teste de transferência consiste em medir, numa sessão de teste pós-treino, sem a plataforma e com duração de 1 minuto, a porcentagem deste tempo que o rato permanece no quadrante-alvo, o quadrante em cujo centro geométrico se encontrava a plataforma submersa nas sessões precedentes de treino. Se um rato efetivamente memorizou onde estava a plataforma, espera-se que ele gaste significativamente mais de 25% do tempo da sessão de teste nadando neste quadrante, a procura da plataforma. A relação entre este parâmetro e a formação de memória espacial é sustentada por estudos farmacológicos e de lesões em áreas cerebrais específicas, como o hipocampo [Redish e Touretzky 1998].

Na sala do LAM do Laboratório de Neuropsicofarmacologia, há uma câmera de vídeo sobre o tanque que permite gravar todo o procedimento. Até o presente momento, o rastreamento dos ratos no LAM tem sido feito de forma manual. O vídeo do rato no tanque é transmitido a um monitor, que tem a sua tela demarcada em quatro quadrantes, através de um risco horizontal e um risco vertical, centrados na tela. Um pesquisador deve assistir ao vídeo do rato no tanque e, utilizando um cronômetro, realiza a medição do tempo de permanência do rato em cada quadrante. Esta é uma tarefa que demanda bastante tempo do pesquisador, além de ser trabalhosa, subjetiva e limitada quanto às possíveis medidas

que podem ser obtidas. Por exemplo, os julgadores (*referees*) de revistas científicas geralmente cobram parâmetros de medida de desempenho do rato durante a sessão de teste no LAM, além da latência e do teste de transferência, tais como [Blokland et al. 2004]: velocidade média de nado (controle da atividade motora do rato), distância percorrida (outra forma de medir a latência), desenho da trajetória de nado, densidade de ocupação de um rato (se o rato permanece mais tempo ou passa várias vezes pelo mesmo local, esta região ficará com cores mais "quentes" do que regiões pouco percorridas, que não são coloridas ou possuem cores mais "frias"), e densidade de ocupação de vários ratos pertencentes a um grupo experimental (somatório da trajetória percorrida de todos os ratos deste grupo, onde regiões mais percorridas por mais ratos possuem cores mais "quentes" do que regiões pouco percorridas, que não são coloridas ou possuem cores mais "frias"). Estes outros parâmetros são muito difíceis, senão impossíveis, de serem obtidos apenas manualmente.

Com o problema entendido, surge a questão: Como automatizar o rastreamento manual no Laboratório de Neuropsicofarmacologia, possibilitando extrair dados quantitativos?

Nesse contexto, as técnicas de Processamento Digital de Imagens (PDI) podem ser utilizadas para automatizar esse processo, bem como permitir a aquisição de parâmetros de desempenho mais complexos. Com o rastreamento implementado em computador, é possível liberar esse tempo de trabalho para que o pesquisador realize outras tarefas mais relevantes em sua pesquisa, além de ser possível obter medidas mais objetivas e variadas. PDI engloba técnicas computacionais que se utilizam de cálculos matemáticos em uma imagem para melhoramento das características visuais ou para extração de dados da mesma [Gonzalez e Woods 2010]. PDI está presente em diversas áreas de conhecimento como a medicina, biologia, geografia, entre outras [Marques Filho e Vieira Neto 1999].

Por se tratar de um problema que possui um nível de complexidade muito grande para ser trabalhado manualmente, a aquisição de informações do LAM demanda a utilização de uma forma automatizada para facilitar o processo. Existem softwares no mercado que realizam a tarefa de monitoramento, porém com alto custo de aquisição. Um exemplo é o EthoVision XT<sup>1</sup> que custa U\$5.850,00 [Noldus 2018].

Devido à impossibilidade de se adquirir um software já disponível no mercado por causa do alto custo, optou-se por desenvolver um software que realize essa tarefa. No projeto de Iniciação Científica (IC) do acadêmico (2017-2018), foi desenvolvido um software de rastreamento de um rato, ou seja, um software que identifica e demarca o caminho percorrido ao longo do tempo em vídeo. Assim, o principal objetivo neste Trabalho de Conclusão de Curso (TCC) foi estudar e adicionar ao software de rastreamento algumas técnicas de extração de características (descritores, medidas) relevantes para os estudos no LAM. Foram implementados neste trabalho: distância percorrida, tempo de percurso, velocidade média, desenho da trajetória de nado e densidade de ocupação de um rato. O software implementado é relevante não só para os pesquisadores do Laboratório de Neuropsicofarmacologia, mas também a outros que trabalhem utilizando o LAM.

Portanto, o objetivo geral deste trabalho é estudar e implementar técnicas para extração de características do caminho percorrido por um rato na tarefa mnemônica do Labirinto Aquático de Morris. Os objetivos específicos são: estudar Processamento Digi-

---

<sup>1</sup><https://www.noldus.com/animal-behavior-research/products/ethovision-xt>

tal de Imagens, estudar técnicas de extração de características e implementar técnicas de extração de características.

O conteúdo deste artigo está organizado nas seguintes seções: Fundamentação Teórica (Seção 2), Materiais e Métodos (Seção 3), Desenvolvimento (Seção 4), Resultados e Discussões (Seção 5), e Considerações Finais (Seção 6).

## 2. Fundamentação Teórica

Nesta seção são abordados os principais conceitos relacionados ao Processamento Digital de Imagens utilizados neste trabalho e sobre os trabalhos correlatos pesquisados.

### 2.1. Processamento Digital de Imagens

O Processamento Digital de Imagens (PDI) engloba técnicas computacionais que se utilizam de cálculos matemáticos em uma imagem para melhoramento das características visuais ou para extração de dados da mesma [Gonzalez e Woods 2010].

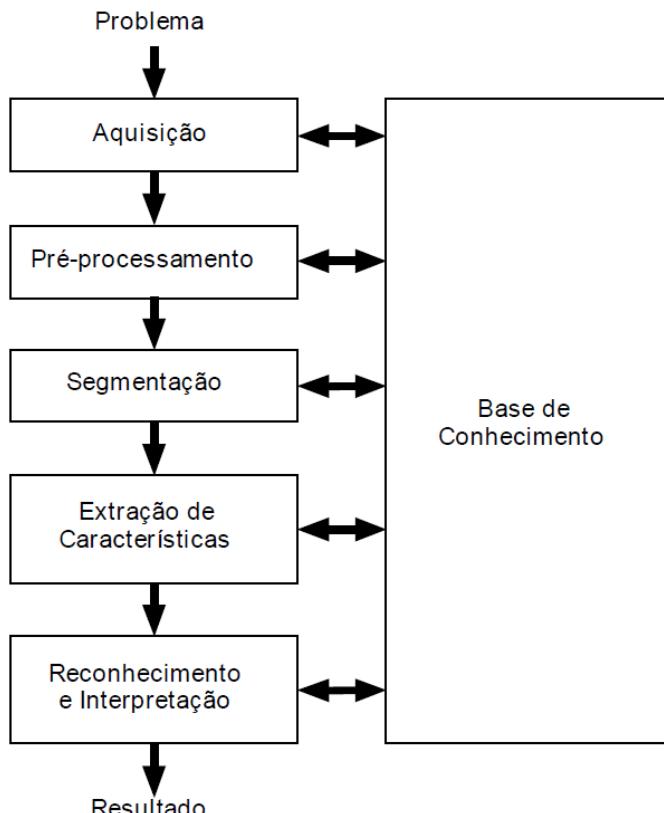
Uma imagem pode ser definida como uma função bidimensional  $f(x, y)$ , na qual  $x$  e  $y$  são coordenadas espaciais e o valor de  $f$  em um par de coordenadas é chamado de intensidade ou nível de cinza da imagem. Uma imagem digital é a representação de uma imagem bidimensional como um conjunto finito de valores chamados de *picture elements (pixels)* [Gonzalez e Woods 2010].

Uma imagem colorida pode possuir diferentes padrões para representar suas cores. Câmeras e monitores normalmente utilizam o padrão RGB (R - Red, G - Green, B - Blue), no qual cada canal necessita de uma função  $f(x, y)$  e possui valores inteiros de 0 até 255 (se a representação for de 8 bits para cada *pixel*, em cada canal). Neste trabalho é utilizado o padrão HSV (H - Hue, S - Saturation, V - Value), que permite separar os componentes: matiz (*Hue*), com valor entre 0 e 360; saturação (*Saturation*), que representa o quanto a cor é dissolvida na luz branca e varia entre 0 e 255; e o último valor sendo a intensidade luminosa da cor (*Value*) variando também de 0 a 255 [Marques Filho e Vieira Neto 1999].

As etapas geralmente mais utilizadas em PDI [Marques Filho e Vieira Neto 1999] podem ser visualizadas na Figura 1 e são explicadas a seguir:

- Aquisição de Imagens: inicialmente é necessário capturar a imagem utilizando algum dispositivo;
- Pré-processamento: a imagem capturada na primeira etapa pode possuir algum tipo de imperfeição, como brilho ou contraste inadequado; para resolver isso, podem ser utilizadas técnicas de aprimoramento da imagem;
- Segmentação: nesta etapa ocorre a busca por objetos de interesse (usando este trabalho como exemplo, o objeto de interesse é o rato, que será procurado nas imagens e destacado);
- Extração de características: a tarefa aqui é tirar informações do objeto que foi segmentado (são as medidas, descritores);
- Reconhecimento e interpretação: na última etapa, a tarefa de reconhecimento consiste em atribuir um rótulo a um objeto com base em suas características (classificação) e a interpretação é a atribuição de significado aos objetos reconhecidos.

- Base de conhecimento: todas as etapas realizadas precisam ter um conhecimento do que ela vai realizar e de que forma, guiando o funcionamento de cada uma. Além disso, a base também pode fazer com que uma etapa seja realizada novamente caso ocorra algum resultado inesperado na mesma.

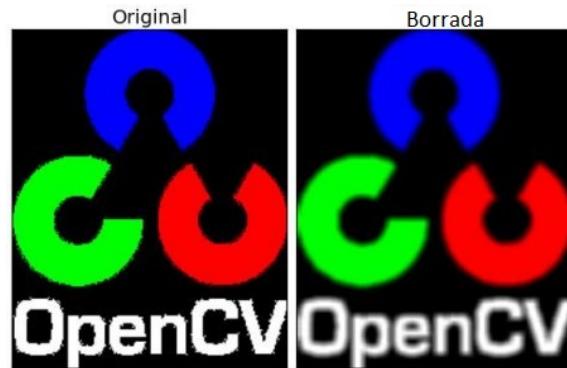


**Figura 1. Etapas em PDI. Fonte: [Marques Filho e Vieira Neto 1999].**

## 2.2. Suavização de imagens

Suavizar uma imagem se trata da aplicação de um filtro para provocar uma suavização na mesma, o que acarreta na remoção de ruídos indesejados que possam atrapalhar no processamento. O operador escolhido para ser utilizado neste trabalho foi o filtro da média, que utiliza uma máscara de tamanho  $N \times N$ . Foi usada uma função da biblioteca OpenCV que implementa esse operador.

No filtro da média a cada iteração, o operador calcula a média dos pixels da imagem que coincidem com a máscara e o resultado é armazenado na imagem resultante [Marques Filho e Vieira Neto 1999]. A escolha de  $N$  é feita com base no grau de borramento desejado. Quanto maior o tamanho da máscara, maior o borramento da imagem resultante. Na Figura 2 está ilustrado um exemplo de suavização utilizando uma máscara  $5 \times 5$ .

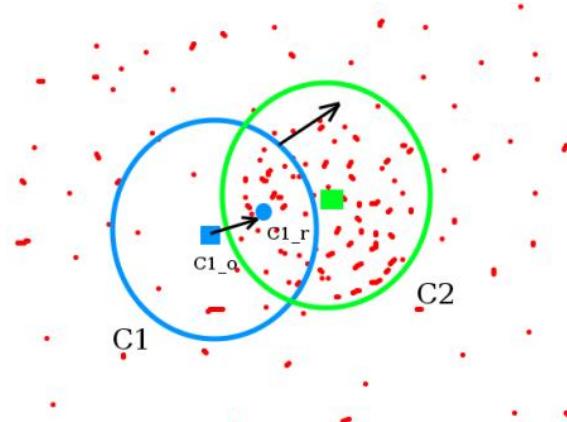


**Figura 2.** Imagem sem filtro (esquerda) e com filtro de suavização (direita). Fonte: [OpenCV 2018d].

### 2.3. Rastreamento de objetos

Rastreamento é o processo que determina a localização de algum objeto ao longo do tempo numa sequência de imagens ou vídeos digitais. Rastrear objetos em vídeo digital é uma tarefa desafiadora, tanto pelas dificuldades em se manipular o vídeo, quanto pelo grau de complexidade que os quadros podem apresentar [Graciano 2007].

Neste trabalho, optou-se por utilizar o algoritmo *Meanshift* disponibilizado na biblioteca OpenCV, pois é um algoritmo eficaz e com bom desempenho de execução. O *Meanshift* é um algoritmo que desloca de forma iterativa uma região de pontos para a maior média de uma região de pontos na sua vizinhança [Montanari 2016]. Na Figura 3 é possível visualizar um exemplo de execução do algoritmo, no qual a primeira iteração ocorre dentro do círculo azul ( $C_1$ ), e a partir do seu centro ( $C_{1\_o}$ ) é calculada a densidade de *pixels* nas regiões vizinhas, obtendo-se outro ponto ( $C_{1\_r}$ ) e movendo o círculo de forma que o novo ponto se torne o centro. Nas próximas iterações o processo se repete novamente, até chegar no ponto verde e obter o círculo verde ( $C_2$ ) que possui a maior densidade de *pixels*.



**Figura 3.** Deslocamento do *Meanshift*. Fonte: [OpenCV 2018b].

### 2.4. Trabalhos Correlatos

Algoritmos para rastreamento de animais já foram desenvolvidos anteriormente. Souza et al. (2008) criou um sistema para rastreamento de múltiplos animais em um vídeo, abor-

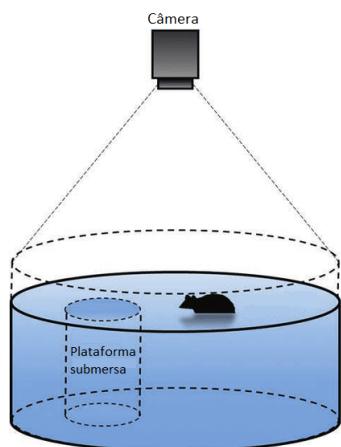
dando estratégias de Computação Natural. O projeto de Chen et al. (2006) utilizou uma FPGA (*Field Programmable Gate Array*) para realizar o processamento, o que faz com que não necessite de um computador. Outro trabalho que realiza o rastreamento de ratos acompanhando a sua cabeça em cada imagem é o de [Hval et al. 2004]. O EthoVision é um sistema comercial que trabalha com rastreamento de vários animais, como ratos e formigas [Noldus 2018].

### 3. Materiais e Métodos

Nesta seção são abordados os materiais e métodos utilizados para desenvolver o trabalho.

#### 3.1. Vídeos

Na Figura 4, pode ser visualizado o esquemático do LAM. Em um tanque preenchido com água, há um rato nadando, procurando a plataforma submersa. Sobre o tanque, há uma câmera gravando o procedimento.



**Figura 4. Labirinto Aquático de Morris. Adaptado de: [Kazantsev et al. 2017].**

Os vídeos foram gravados na sala do LAM, do Laboratório de Neuropsicofarmacologia, que cedeu as imagens para utilização neste trabalho. Na Figura 5, pode-se visualizar o tanque, que possui 1,70 m de diâmetro. A câmera é mostrada na Figura 6.

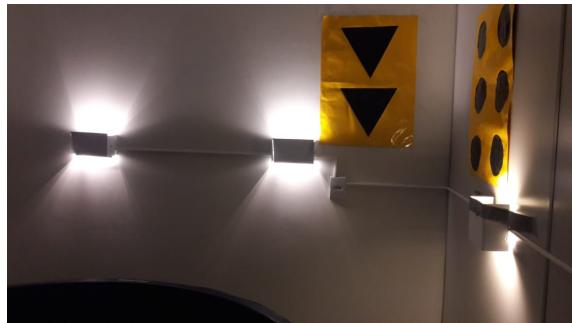


**Figura 5. O tanque do LAM no Laboratório de Neuropsicofarmacologia.**



**Figura 6. A câmera do LAM no Laboratório de Neuropsicofarmacologia.**

Em volta do tanque existem pistas espaciais fixadas nas paredes para servir como guia para os ratos. Essas pistas espaciais são desenhos de figuras geométricas e lâmpadas nas paredes. Através dessas pistas, o rato consegue se guiar, fazendo triangulação de posição. Na Figura 7, pode-se visualizar algumas dessas pistas.



**Figura 7. Pistas espaciais externas ao tanque.**

Na Figura 8, pode-se visualizar um *frame*<sup>2</sup> em que aparece um rato nadando no tanque em busca da plataforma. Já na Figura 9, é possível ver o rato repousando sobre a plataforma encontrada.

Os vídeos precisaram ser editados para remover cenas em que não é possível realizar análise, como momentos que o tanque está vazio e cenas em que o pesquisador aparece sobre o tanque. Além disso, devido à limitação da altura do teto, não foi possível abranger todo o tanque nas imagens, ficando cortadas pequenas porções do tanque nas partes superior e inferior dos vídeos, conforme pode ser visto nas Figuras 8 e 9. Para os testes neste trabalho, foi selecionado um trecho de vídeo em que o rato realiza todo seu percurso aparecendo no vídeo, ou seja, sem passar pelas regiões cortadas do vídeo.

---

<sup>2</sup>Quadro de vídeo.



**Figura 8. Rato nadando, procurando a plataforma.**



**Figura 9. Rato em repouso sobre a plataforma encontrada.**

### 3.2. Ambiente computacional de desenvolvimento

Os recursos tecnológicos utilizados foram um Notebook ASUS com processador Intel Core i5, 8 GB de RAM, Windows 10 e acesso à internet. A programação do software foi realizada no ambiente de desenvolvimento Visual Studio, utilizando a linguagem de programação C++ e a biblioteca *open source* OpenCV.

O Visual Studio é um ambiente de desenvolvimento integrado (IDE – *Integrated Development Environment*), desenvolvido pela Microsoft com suporte a desenvolvimento *mobile*, web e plataforma Windows [Microsoft 2018]. A versão utilizada foi a Visual Studio 2017 Community, disponível para download gratuito no site oficial da Microsoft<sup>3</sup>.

O OpenCV<sup>4</sup> (*Open Source Computer Vision Library*) é uma biblioteca totalmente livre, para uso acadêmico e comercial no desenvolvimento de aplicativos de Visão Computacional. A biblioteca possui compatibilidade com aplicações *desktop* e *mobile*, com suporte para as linguagens de programação C++, Java e Python, além de possuir mais de 500 algoritmos para utilização [OpenCV 2018c]. A versão utilizada foi a 3.4.0.

### 3.3. Método de desenvolvimento

Foram seguidas as etapas de PDI, conforme descrito na Seção 2.1. A aquisição das imagens foi realizada por pesquisadores do Laboratório de Neuropsicofarmacologia, que geraram vídeos de ratos nadando no LAM. As etapas de pré-processamento e segmentação foram abordadas no projeto de IC do acadêmico, através da utilização do filtro de média e *Meanshift* do OpenCV para localização e armazenamento da posição do rato em cada

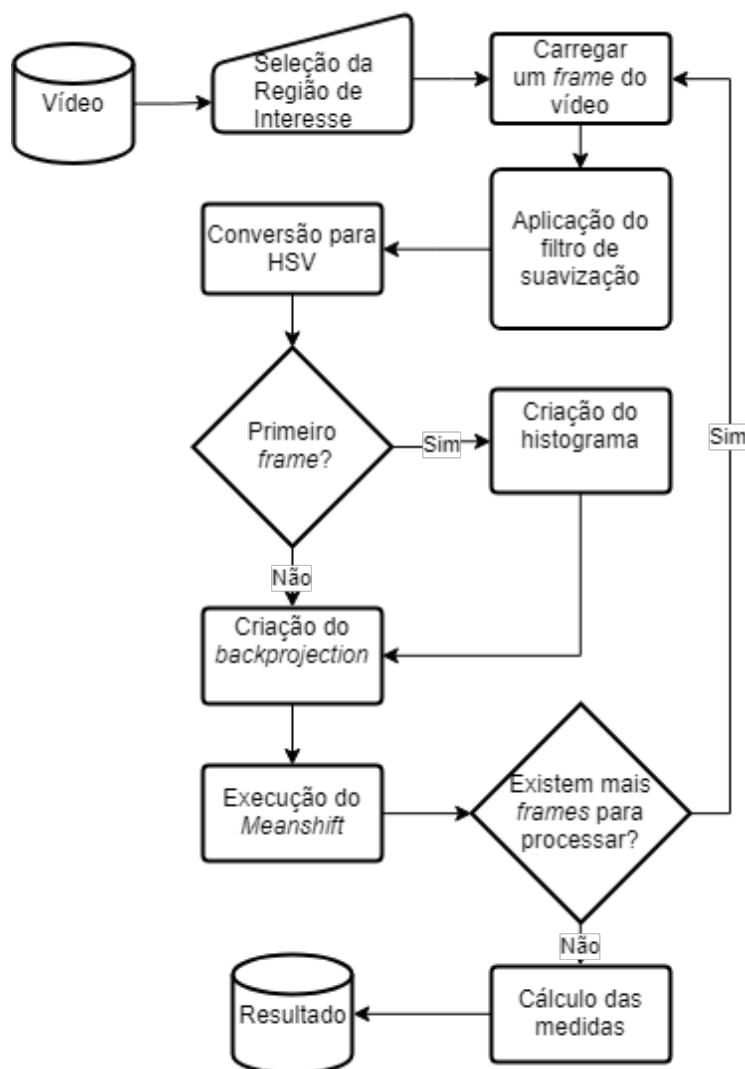
<sup>3</sup><https://www.visualstudio.com/pt-br/vs/>

<sup>4</sup><https://opencv.org/>

*frame* do vídeo. A etapa de extração de características foi desenvolvida neste trabalho. Reconhecimento e interpretação não se aplicam a este trabalho, pois o objeto de interesse nas imagens já é conhecido (rato), não necessitando ser classificado. A análise das medidas obtidas é de responsabilidade dos pesquisadores do Laboratório de Neuropsicofarmacologia, que depende dos experimentos sendo realizados com os ratos (trabalhos futuros).

#### 4. Desenvolvimento

O software desenvolvido permite tanto execução em tempo real de vídeos diretamente adquiridos por uma câmera, quanto processamento de arquivos de vídeo. Para facilitar os testes, neste trabalho foram utilizados apenas vídeos previamente gravados. Todo o processo de execução está resumido no fluxograma na Figura 10, com todas as etapas sendo explicadas logo a seguir.



**Figura 10. Fluxograma de execução do software.**

Primeiro, um vídeo é carregado para processamento ou inicia-se a aquisição em tempo real. Em seguida, deve-se selecionar a região onde o rato se encontra. As

informações dos pixels dessa região de interesse selecionada são armazenados para posteriormente serem utilizadas na busca pelo rato nos *frames* seguintes.

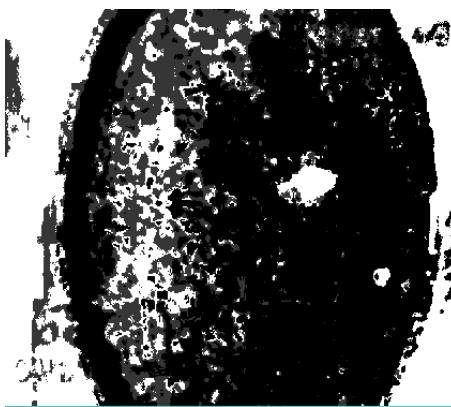
Para cada *frame* do vídeo, o processamento se inicia com a aplicação do filtro de suavização, cujo propósito é reduzir ruídos das imagens que poderiam prejudicar a detecção do rato. A seguir, o *frame* é convertido do padrão de cores RGB para o padrão de cores HSV.

Para realizar a etapa de segmentação, foi criada uma máscara binária (Figura 11) que possui o valor 1 para as regiões com cores similares ao rato e 0 para regiões que não se assemelham. Ainda na etapa de segmentação, é separado o primeiro componente de matiz (*Hue*) para ser utilizado na criação do histograma, que ocorre na primeira iteração, ou seja, o primeiro *frame* analisado. Histograma é um conjunto de números que indicam a quantidade de pixels de cada cor presente em uma imagem, normalmente representado por um gráfico de barras [Gonzalez e Woods 2010]. A máscara criada anteriormente também auxilia na criação do histograma, fazendo com que as regiões onde não possuem cores similares ao rato não sejam calculadas durante o processo de criação do mesmo.



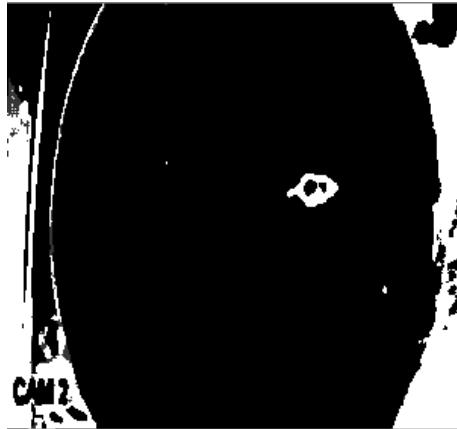
**Figura 11. Máscara binária resultante.**

Com o uso do histograma foi obtido o *Backprojection* (Figura 12), imagem binária utilizada para verificar quais regiões de alguma outra imagem possuem características semelhantes a si [OpenCV 2018a].



**Figura 12. Backprojection resultante.**

Entretanto, o *Backprojection* apresentou grandes quantidades de ruídos ainda, então foi realizada a aplicação do operador *AND* com a máscara. O resultado é ilustrado na Figura 13.



**Figura 13.** *Backprojection* após aplicar o operador *AND* utilizando a máscara criada.

Com todos os dados obtidos, o *Meanshift* utiliza o *Backprojection* e a região de interesse para fazer o processo explicado na Seção 2.3. Após o algoritmo ter convergido, é possível utilizar o centro de massa da região analisada para obter o ponto da localização atual do rato, que é salvo em uma lista de posições.

A obtenção do tempo é dada pelos segundos em que o rato estava em movimento no vídeo. Para a distância, foi realizado o cálculo da distância euclidiana entre os pontos armazenados na lista de posições obtidos de cada *frame*, conforme mostrado na Equação 1, na qual  $d_{12}$  é a distância calculada entre os pontos  $(x_1, y_1)$  e  $(x_2, y_2)$ . A distância é calculada entre o 1º e o 2º pontos, em seguida entre o 2º e o 3º pontos, e assim sucessivamente com todos os pontos disponíveis. A seguir, essas distâncias entre pontos são somadas para se obter a distância completa percorrida pelo rato.

$$d_{12} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (1)$$

Com auxílio do GIMP<sup>5</sup> foi possível medir o diâmetro do tanque como sendo de 340 pixels. Com essa medida, foi possível utilizar uma regra de três para calcular que o diâmetro de um pixel é aproximadamente 5 mm (milímetros) de comprimento, conforme cálculo ilustrado na Equação 2.

$$\frac{340 \text{ pixels}}{1 \text{ pixel}} = \frac{1700 \text{ mm}}{X \text{ mm}} \quad (2)$$

$$X = 1700 / 340 = 5$$

Dessa forma, foi possível obter a distância que o rato percorreu, multiplicando a distância total pelo diâmetro de um pixel. Com a distância e o tempo, foi possível calcular

---

<sup>5</sup>GNU Image Manipulation Program - <https://www.gimp.org/>

a velocidade média, conforme Equação 3

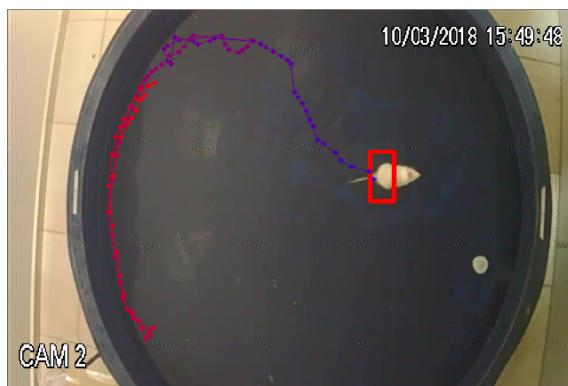
$$V_m = \frac{\text{Distância percorrida}}{\text{Tempo de nado}} \quad (3)$$

Para obter a densidade de ocupação, os *frames* são convertidos para escala de cinza. A partir do segundo *frame*, é realizada uma subtração dele com o anterior para obter uma máscara com o rato destacado. Essa máscara é transformada em uma imagem binária e seus valores são somados à uma imagem que inicialmente foi preenchida com zeros. Então, é aplicada uma coloração na imagem acumulada em níveis de cinza para obter a densidade de ocupação.

Foram inseridos no programa atalhos de teclado para pausar a execução, mostrar o caminho percorrido em cima da imagem do rato e mostrar o *backprojection* da imagem. Foram utilizadas telas separadas para mostrar o histograma, o caminho percorrido e a densidade de ocupação.

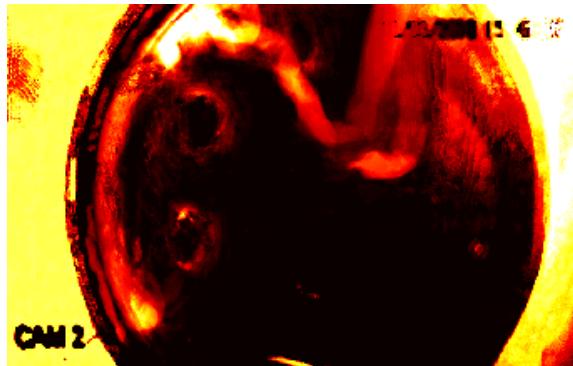
## 5. Resultados e Discussões

O rastreamento do rato e a extração dos dados quantitativos são feitos de forma automática, sendo apenas necessário que o usuário que está interagindo com o software insira a posição inicial do rato. Na Figura 14 é demonstrado um exemplo de resultado obtido para o desenho da trajetória de nado. Foi utilizada a cor vermelha no inicio do percurso que vai se alterando para a cor azul para identificação do ponto de inicio e fim.

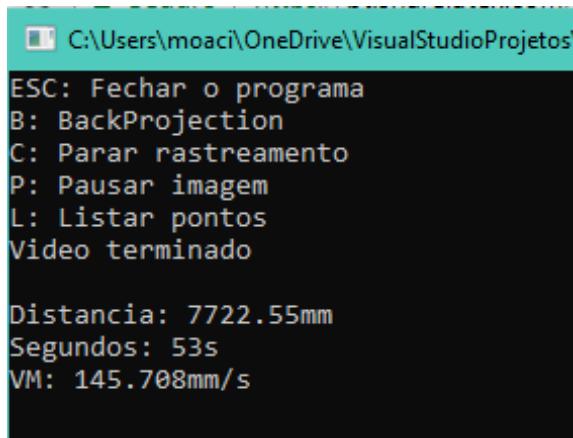


**Figura 14. Resultado do caminho tracejado pelo software após analisar um vídeo.**

Na Figura 15, pode-se visualizar o resultado da avaliação de densidade de ocupação de um rato. As regiões mais claras evidenciam os locais em que o rato permaneceu por mais tempo. A distância percorrida, tempo de nado e velocidade média são mostrados no *prompt* de comando do aplicativo, como pode ser visto na Figura 16.



**Figura 15.** Resultado do mapa de densidade criado pelo software após analisar um vídeo.



**Figura 16.** Resultados após o software analisar um vídeo.

## 6. Considerações Finais

Realizar a análise do caminho percorrido pelo rato no Labirinto Aquático de Morris de forma manual é uma tarefa que demanda muito tempo, além de ser trabalhosa e cansativa. Quando é necessário extrair informações dessa análise o problema se torna ainda maior, devido às limitações e dificuldades de extração de algumas medidas, como por exemplo o mapa de densidade de ocupação.

Com o software desenvolvido para o rastreamento e a extração de dados, o processo se torna muito mais fácil, possibilitando ao pesquisador realizar outras tarefas mais revelantes em sua pesquisa, além de ser possível obter medidas mais complexas, mais objetivas e variadas.

Para desenvolver o programa foram pesquisados os conceitos de rastreamento de objetos e a utilização da técnica *Meanshift* de processamento em tempo real. Durante o desenvolvimento, uma dificuldade foram os ruídos presentes nos vídeos, que dificultaram a execução do algoritmo. A solução para o problema foi a utilização da técnica de suavização de imagem. Outra dificuldade foram as bordas superior e inferior do tanque estarem cortadas no vídeo. Quando o rato some nessas regiões, o algoritmo geralmente não consegue encontrá-lo mais, sendo necessário o usuário selecionar novamente a posição

do rato quando ele reaparecer no vídeo. Para contornar este problema, foram utilizados somente trechos de vídeo em que o rato não vai para essas regiões não abrangidas pela câmera.

Portanto, como trabalhos futuros propõe-se a implementação de tratamento do caso quando o rato entra em pontos cegos do vídeo sem que seja necessário o reposicionamento manual do objeto de interesse (rato). Propõe-se também a implementação da extração de outras medidas e de uma Interface Gráfica para o Usuário (GUI – *Graphical User Interface*), a fim de tornar a utilização do software mais intuitiva.

## Agradecimentos

Agradecemos aos pesquisadores do Laboratório de Neuropsicofarmacologia pelos vídeos e esclarecimento de dúvidas sobre o problema em estudo.

## Referências

- Blokland, A., Geraerts, E., e Been, M. (2004). A detailed analysis of rats' spatial memory in a probe trial of a morris task. *Behavioural brain research*, 154(1):71–75.
- Bures, J., Fenton, A., Kaminsky, Y., e Zinyuk, L. (1997). Place cells and place navigation. *Proceedings of the National Academy of Sciences*, 94(1):343–350.
- Chen, Y.-J., Li, Y.-C., Huang, K.-N., e Young, M.-S. (2006). The implementation of a stand-alone video tracking and analysis system for animal behavior measurement in morris water maze. In *Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference of the*, pages 1766–1768. IEEE.
- De Hoz, L., Martin, S. J., e Morris, R. G. (2004). Forgetting, reminding, and remembering: the retrieval of lost spatial memory. *PLoS biology*, 2(8):e225.
- Gonzalez, R. C. e Woods, R. C. (2010). *Processamento digital de imagens*. Pearson Educación.
- Graciano, A. B. V. (2007). Rastreamento de objetos baseado em reconhecimento estrutural de padroes. *Universidade de São Paulo, São Paulo, Brasil*.
- Hval, E. S., Barrett, S. F., Wilcox, M., e Bailey, T. A. (2004). An improved morris water maze tracking algorithm for psychophysical studies. *Annals of biomedical engineering*, 32(8):1143–1154.
- Kazantsev, A., Mironov, A., Tarasov, R., Shabaev, A., Firsov, N., e Ruban, E. (2017). Open surgery for treating multiple aneurysms of intracranial cerebral arteries. *Fundamental and Clinical Medicine*, 2:100–106.
- Marques Filho, O. e Vieira Neto, H. (1999). *Processamento digital de imagens*. Brasport.
- Microsoft (2018). *Visual Studio* Em: <<https://www.visualstudio.com/pt-br/vs/>>. Acesso em: 31 de maio de 2018.
- Montanari, R. (2016). *Detecção e classificação de objetos em imagens para rastreamento de veículos*. PhD thesis, Universidade de São Paulo.
- Morris, R. G. M. (1984). Developments of a water-maze procedure for studying spatial learning in the rat. *Journal of Neuroscience Methods*, 11(1):47–60.

- Morris, R. G. M. (1989). Synaptic plasticity and learning: Selective impairment of learning in rats and blockade of long-term potentiation in vivo by the n-methyl-d-aspartate receptor antagonist ap5. *The Journal of Neuroscience*, 9(9):3040–3057.
- Noldus (2018). *EthoVision XT* Em: <<https://www.noldus.com/animal-behavior-research/products/ethovision-xt>>. Acesso em: 31 de maio de 2018.
- OpenCV (2018a). *Back Projection* Em: <[https://docs.opencv.org/3.0-beta/doc/tutorials/imgproc/histograms/back\\_projection/back\\_projection.html](https://docs.opencv.org/3.0-beta/doc/tutorials/imgproc/histograms/back_projection/back_projection.html)>. Acesso em: 31 de maio de 2018.
- OpenCV (2018b). *Meanshift and Camshift* Em: <[https://docs.opencv.org/3.4.0/db/df8/tutorial\\_py\\_meanshift.html](https://docs.opencv.org/3.4.0/db/df8/tutorial_py_meanshift.html)>. Acesso em: 31 de maio de 2018.
- OpenCV (2018c). *OpenCV* Em: <<https://opencv.org/>>. Acesso em: 31 de maio de 2018.
- OpenCV (2018d). *Smoothing Images* Em: <[https://docs.opencv.org/3.4.0/d4/d13/tutorial\\_py\\_filtering.html](https://docs.opencv.org/3.4.0/d4/d13/tutorial_py_filtering.html)>. Acesso em: 31 de maio de 2018.
- Redish, A. D. e Touretzky, D. S. (1998). The role of the hippocampus in solving the morris water maze. *Neural computation*, 10(1):73–111.
- Schimanski, L. e Nguyen, P. (2004). Multidisciplinary approaches for investigating the mechanisms of hippocampus-dependent memory: a focus on inbred mouse strains. *Neuroscience & Biobehavioral Reviews*, 28(5):463–483.
- Souza, R. H. C. d. (2008). Rastreamento de animais por imagens de vídeo em experimentos de laboratório. Master's thesis, Universidade Estadual de Campinas (UNICAMP), Campinas, SP.
- Spink, A. J., Tegelenbosch, R. A. J., Buma, M. O. S., e Noldus, L. P. J. J. (2001). The ethovision video tracking system: a tool for behavioral phenotyping of transgenic mice. *Physiology & Behavior*, 73(5):731–744.
- Stewart, C. e Morris, R. (1993). The watermaze. *Behavioural neuroscience*, 1:107–122.