



Universidade Católica Dom Bosco
Curso de Bacharelado em Engenharia de Computação

Desenvolvimento de um Sistema de Visão
Computacional para o Monitoramento e Controle de
Experimentos Laboratoriais Realizados com Larvas do
Aedes aegypti L.

Jaime Henrique Ferreira de Sá Queiroz

Prof. Orientador: Dr. Hemerson Pistori

*Relatório Final submetido como um dos requisitos
para a obtenção do título de Bacharel em Engenharia
de Computação.*

UCDB - Campo Grande - MS - Novembro/2010

à minha família e amigos.

Agradecimentos

Agradeço especialmente ao meu pai e a minha mãe por estarem sempre ao meu lado e por propiciarem o meu ingresso em um curso de formação superior que pude escolher. Agradeço a paciência dos meus irmãos por roubar toda a banda de internet quando eu estava em casa. Agradeço aos meus amigos e colegas de classe que conheci durante esses anos de graduação pelas conversas, grupos de estudos e momentos de lazer. Agradeço ao pessoal do GPEC/INOVISAO, mesmo aqueles que já saíram, pelo cohecimento que me foi passado nesses anos de pesquisa. Um especial agradecimento também ao meu orientador Hemerson Pistori pela paciência e compreensão, por passar muitos conhecimentos, ter influenciado na minha formação profissional e além de ter sido um ótimo professor, ter sido um grande amigo. Bom, a todos aqueles que de alguma forma estiveram comigo durante essa jornada acadêmica meus mais sinceros, muito obrigado!

Resumo

A importância do estudo e desenvolvimento de novas formas de se combater a proliferação do mosquito transmissor da Dengue torna-se cada vez mais evidente diante dos repetidos surtos desta doença no Brasil, especialmente no estado do Mato Grosso do Sul. Este trabalho propõe o desenvolvimento de um sistema de visão computacional que irá tornar mais confiável, eficiente e reprodutível o teste de potenciais larvicidas para as larvas do vetor da Dengue, o *Aedes aegypti*. Atualmente, estes testes, que demandam 24 horas de observação contínua, são realizados por humanos, que acabam sendo submetidos a condições de grande estresse pela realização de um trabalho repetitivo e que demanda muita atenção. Este sistema de visão computacional permitirá também a redução no tempo necessário para se chegar a novos larvicidas, pois uma quantidade maior de testes poderá ser realizada.

Palavras-chave: Visão computacional, rastreamento, segmentação, extração de atributos, dengue.

Abstract

The importance of studying and developing new ways to combat the proliferation of mosquito that transmits Dengue becomes increasingly evident before the repeated outbreaks of this disease in Brazil, especially in Mato Grosso do Sul. This paper proposes the development of a computer vision system that will make it more reliable, efficient and reproducible testing of potential larvicides for the larvae of the dengue vector, the *Aedes aegypti*. Currently, these tests, which require 24 hours of continuous observation, are performed by humans, who end up being subjected to conditions of great stress for performing repetitive work that demands attention. This computer vision system will also reduce the time needed to reach new larvicides, because more tests can be performed.

Keywords: Computer vision, tracking, segmentation, extraction of attributes, dengue.

Conteúdo

1	Introdução	9
2	Fundamentação Teórica	11
2.1	Visão Computacional	11
2.1.1	Pré-processamento	11
2.1.2	Segmentação	13
2.1.3	Rastreamento	15
2.1.4	Extração de Atributos	17
2.1.5	Reconhecimento de Padrões	19
3	Desenvolvimento	20
3.1	Módulo de Rastreamento de Múltiplas Larvas	20
3.2	Módulo de Classificação Automática	21
3.3	Módulo de Estatística	23
3.4	Integração dos Módulos em um Sistema de Informação	23
4	Experimentos	26
5	Considerações Finais	29

Lista de Figuras

2.1	Vizinhança 3x3 ao redor do ponto de referência.	12
2.2	(1)Imagem original, (2) imagem resultante.	12
2.3	Sequência de imagens originais.	14
2.4	Sequência de imagens segmentadas usando subtração de fundo.	14
2.5	Exemplo de rastreamento utilizando filtro de partículas.	16
3.1	Sequência de imagens onde irá ocorrer oclusão	21
3.2	Estados de Movimentação da Larva	21
3.3	Diagrama da primeira etapa.	22
3.4	Telas do módulo de estatística.	24
3.5	Diagrama entidade-relacionamento.	24
3.6	Tela inicial.	25

Lista de Tabelas

4.1	Taxa de acerto dos extratores.	27
4.2	Média da taxa acerto para cada 10 sequências de observações de tamanhos diferentes.	28

Capítulo 1

Introdução

Os recentes casos de epidemia de Dengue em várias cidades do estado do Mato Grosso do Sul, além de outros estados, despertaram várias questões em relação à eficácia das campanhas de combate à doença e também aos métodos preventivos. Por exemplo, a região Centro-Oeste finalizou o ano de 2009 com uma taxa de incidência de 857,6 casos para cada 100.000 habitantes (alta incidência) [inf09].

Um grupo de pesquisadores na Univesidade Católica Dom Bosco desenvolvem e avaliam a eficácia de larvicidas extraídos de plantas contra as larvas do mosquito *Aedes aegypti*. Testes realizados para o desenvolvimento de novos larvicidas implicam análise da taxa de mortalidade de larvas graduando-se as concentrações do larvicida em teste. A análise da mortalidade é feita através da observação visual de um recipiente contendo as larvas em meio líquido com a substância larvicida durante um período de 24 horas. Um especialista periodicamente observa os recipientes e realiza a identificação e o registro de larvas mortas e vivas. Terminado o período de análise e com base nos registros periódicos chega-se a um taxa de mortalidade das larvas sob efeito da substância. De posse desses dados e utilizando um conjunto de métodos estatísticos, chega-se a uma concentração letal do larvicida.

Como esses testes são realizados por seres humanos e por tratar-se de uma tarefa que demanda muita atenção do observador, os registros das observações podem estar sujeitas a erros. Outro fator que contribui para o aumento de erros é a exaustão que o observador sofre devido ao longo período de observação. A subjetividade também está presente durante a observação, pois a identificação do estado atual da larva não é uma tarefa simples devido ao movimento característico da larva sofrer algumas alterações quando ela esta morrendo.

Por causa desses fatores, um sistema que realize automaticamente a identificação do estado vital da larva em instantes definidos é importante nos

testes realizados para o desenvolvimento de novos larvicidas. O emprego da visão computacional é uma alternativa para eliminar os fatores humanos que ocasionam erros durante os testes. Assim a implementação de um sistema de visão computacional que substitua a presença humana durante o período de observação é o objetivo do projeto LARVIC, criado pelo grupo de pesquisa INOVISAO.

No projeto LARVIC, determinar a taxa de mortalidade está sendo automatizada da seguinte maneira. Primeiramente, uma câmera é posicionada acima dos recipientes onde estão contidas as larvas em meio líquido. Posteriormente, as imagens adquiridas pela câmera são segmentadas, gerando assim imagens com apenas os objetos de interesse que no caso são as larvas. Essas imagens são passadas a um filtro preditivo que irá realizar o rastreamento onde será obtida a localização de cada larva. Com cada larva devidamente localizada utiliza-se um extrator de atributos baseado em forma juntamente com um classificador onde obtem-se uma posição característica da larva. Depois de repetidas as etapas anteriores em um determinado número de imagens é criada uma sequência de observação que com base nelas irá gerar atributos temporais para outro classificador que irá determinar o estado vital da larva.

No próximo capítulo serão explanados os algoritmos e métodos utilizados nas etapas de segmentação, rastreamento e extração de atributos.

Capítulo 2

Fundamentação Teórica

2.1 Visão Computacional

A visão computacional pode ser definida como um conjunto de técnicas e métodos computacionais que substitui o sentido visual humano e da capacidade de interpretação dos seres humanos a partir de dispositivos de captura de imagens. Na indústria, a visão computacional basicamente utiliza dispositivos óticos para capturar e interpretar de maneira automática uma imagem real para obter informações ou controlar processos e máquinas.

Um sistema de visão computacional pode ser dividido em três níveis: baixo-nível, nível-médio e alto-nível. Os processos de baixo-nível envolvem operações como redução de ruído ou melhorias no contraste de uma imagem. Os processos de nível-médio são operações do tipo segmentação e reconhecimento de objetos na imagem. Os processos de alto-nível estão relacionados com as tarefas de cognição associadas com a visão humana [GWE06].

2.1.1 Pré-processamento

O principal objetivo desta etapa é processar uma imagem de modo que a imagem resultante seja mais adequada, que a imagem original, para a aplicação desejada [eNVH99]. No desenvolvimento do sistema de visão computacional desse trabalho esta etapa é responsável pela transformação da imagem para tons de cinza e redução de ruídos na imagens através de umas das técnicas de filtragem no domínio espacial.

As técnicas de filtragem no domínio espacial são aquelas que atuam diretamente sobre a matriz de pixels gerada pela imagem digitalizada. Logo, as funções de processamento de imagens no domínio espacial podem ser expressas pela equação 2.1:

$$g(x, y) = T[f(x, y)] \quad (2.1)$$

onde: $g(x, y)$ é a imagem processada, $f(x, y)$ é a imagem original e T é um operador em f , definido em uma certa vizinhança de (x, y) .

A vizinhança definida ao redor de (x, y) é a vizinhança de 8 conectivos do pixel de referência, o que equivale a uma região 3x3, na qual o pixel central é o de referência como indica a Figura 2.1 [EST09].

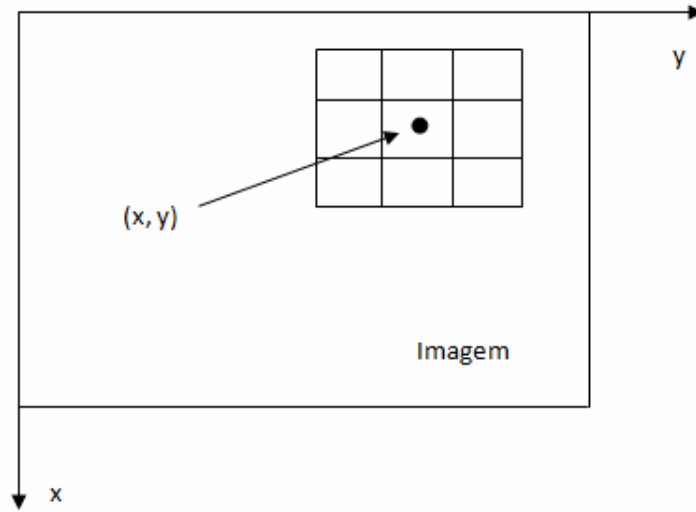
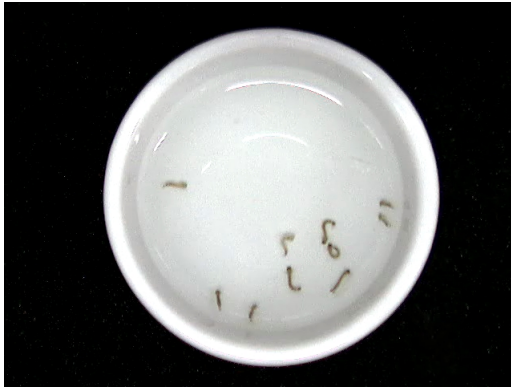


Figura 2.1: Vizinhança 3x3 ao redor do ponto de referência.



(1)



(2)

Figura 2.2: (1) Imagem original, (2) imagem resultante.

A técnica de filtragem no domínio espacial utilizada nesta etapa foi o filtro de mediana. Nesta técnica, o nível de cinza do pixel referencial é substituído pela mediana dos pixels situados em sua vizinhança. A mediana m de um conjunto de n elementos é um valor tal que metade dos n elementos do conjunto situa-se abaixo de m , e a outra metade acima de m . Quando n é ímpar, a mediana é calculada pela média aritmética dos dois elementos mais próximos do centro [GW00]. Na Figura 2.2 pode-se observar a aplicação do filtro de mediana em uma imagem.

2.1.2 Segmentação

O primeiro passo na análise de imagens é a segmentação que consiste em usar o computador para definir na imagem recortes automáticos ao redor de objetos de interesse. A segmentação subdivide uma imagem em suas partes ou objetos constituintes. O nível até o qual essa subdivisão deve ser realizada, assim como a técnica utilizada, depende do problema que está sendo resolvido [GW00]. Neste trabalho será abordada a técnica de subtração de fundo, devido aos experimentos possuírem um fundo estático que favorece a utilização dessa técnica.

A vantagem de se utilizar a técnica de subtração de fundo é pelo seu custo de processamento ser baixo e por ser de fácil implementação. Essa técnica consiste no cálculo da diferença simétrica entre cada pixel do quadro atual e do modelo de fundo. Se essa diferença for maior que um limiar, este pixel é considerado como pertencente ao fundo. Para estimar o modelo de fundo foi adotada uma solução trivial que é filmar o cenário sem os objetos de interesse. A desvantagem desse método é a inadaptabilidade com relação as modificações no fundo da cena, como por exemplo a mudança de iluminação ou objetos sem relevância que entram em cena e em seguida ficam estáticos. Como pode ser observado na sequência de imagens antes da segmentação na Figura 2.3 e respectivamente logo depois foi efetuado a subtração de fundo na Figura 2.4.

A imagem segmentada através da subtração de fundo é obtida pela equação 2.2, onde $Bn(x)$ é a intensidade do modelo de fundo para o pixel x , $Tn(x)$ é o limiar pré-definido atribuído ao pixel x , $In(x)$ é a intensidade do quadro que se deseja segmentar e n é a quantidade de quadros capturados até o momento. O alvo $An(x)$ estimado durante o processamento do quadro n para o pixel x é calculado como a diferença simétrica maior que um limiar $Tn(x)$ entre o quadro atual e o modelo de fundo estimado.

$$An(x) = |In(x) - Bn(x)| > Tn(x) \quad (2.2)$$

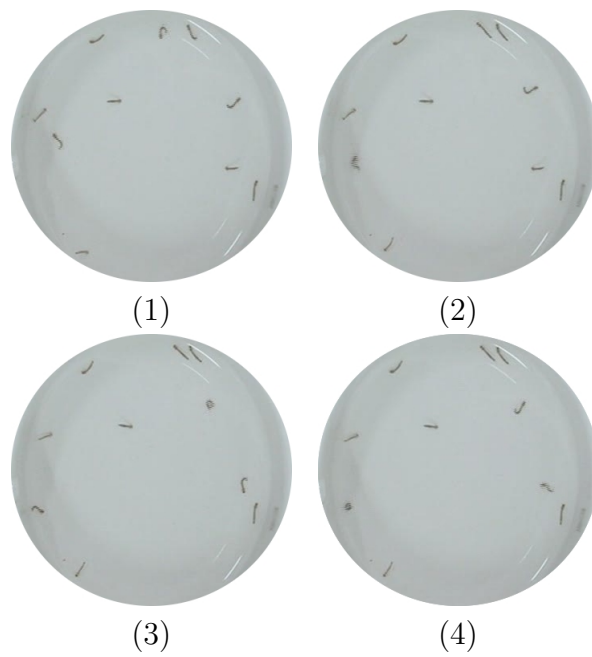


Figura 2.3: Sequência de imagens originais.

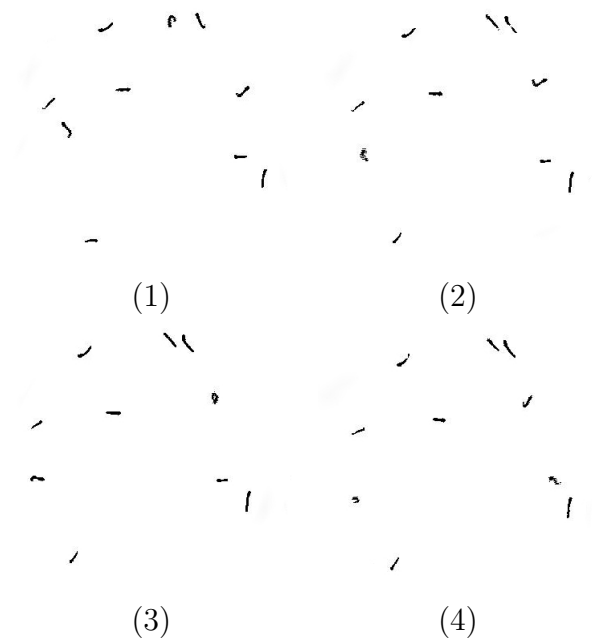


Figura 2.4: Sequência de imagens segmentadas usando subtração de fundo.

2.1.3 Rastreamento

O rastreamento de múltiplos objetos pode ser interpretado como um problema de estimação de estados de um número desconhecido de alvos que se movimentam em um sistema dinâmico [HLCP00] e com a utilização de filtros preditivos é possível estimar o estado ótimo do sistema. Para isso, deve-se utilizar um modelo matemático da dinâmica do sistema para propagar os estados e depois, combinar o estado estimado com a probabilidade dos estados realmente terem ocorrido [GOL04]. Existem diversos filtros preditivos, como o filtro de Kalman e o filtro de partículas, que são apropriados para diferentes tipos de modelagens do sistema. Os filtros preditivos são particularmente interessantes quando se trabalha com dados corrompidos e ruído, como por exemplo, durante o rastreamento de objetos que são constantemente ocultos uns pelos outros.

O filtro de partículas, um tipo de filtro preditivo, pode ser aplicado em problemas envolvendo modelos não-lineares e não-Gaussianos. Para a solução de rastreamento de um objeto é dividido em três etapas: a predição, a observação e a correção [GOL04, HLCP00].

Na predição, um modelo de dinâmica é aplicado a cada uma das partículas, para se calcularem seus novos valores, com base em seus valores anteriores. Na observação, cada partícula recebe um peso com base nas informações obtidas em um modelo de observação e por atributos extraídos a partir da imagem corrente, sendo esse peso proporcional a sua proximidade com os atributos extraídos das imagens. Na correção (*resampling*), uma nova lista de partículas é gerada a partir de uma lista anterior, levando em consideração os pesos, de forma que partículas com maior peso tenham maior probabilidade de serem escolhidas para participar da nova lista [Rek04].

Para exemplificar, considere o rastreamento de larvas em que o objetivo é identificar a coordenada em um plano bidimensional referente ao centro de massa de cada objeto/larva. Na Figura 2.5, temos a ilustração passo a passo do rastreamento da larva utilizando um filtro de partícula com 12 partículas.

A Figura 2.5(a) ilustra a predição realizada pelo filtro para o próximo estado do sistema. No exemplo, o filtro faz uma predição das coordenadas do objeto na próxima imagem da sequência. Essa predição é realizada com base nas ocorrências do passado – por exemplo, os valores dos centros de massa do objeto nos quadros anteriores – e no modelo de dinâmica, que define como as variações no estado do sistema ocorrem.

Em seguida, o sistema faz uma observação do estado atual do sistema. No nosso exemplo, o sistema verifica qual a coordenada real (ou aproximada) do centro de massa da larva. Essa observação é representada na Figura 2.5(b), em que a elipse representa graficamente o centro de massa calculado

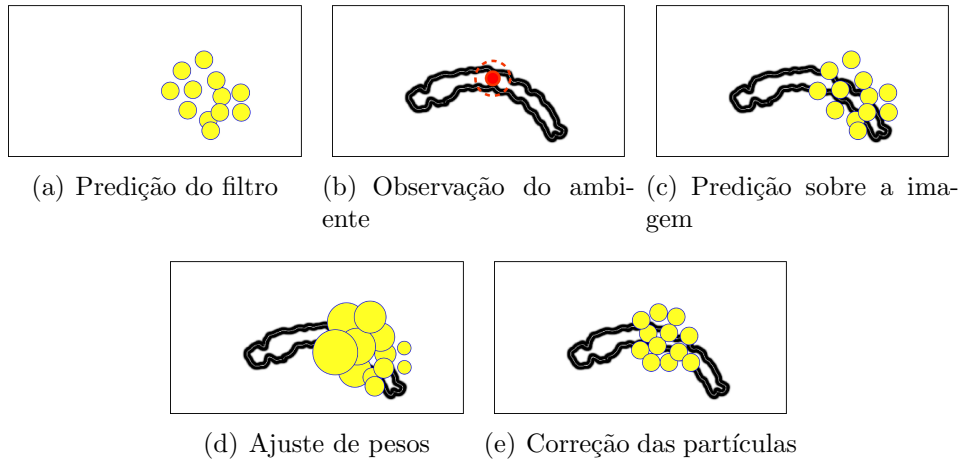


Figura 2.5: Exemplo de rastreamento utilizando filtro de partículas.

na imagem observada. Já a Figura 2.5(c) exibe a predição realizada no passo anterior sobre a larva obtida na observação, apenas para que o leitor possa verificar a distância entre o centro de massa real e os centros de massa de cada partícula da predição do filtro.

Dados os valores reais do estado do sistema, o filtro realiza um ajuste nos pesos das partículas, com o intuito de melhorar a precisão das próximas previsões. De acordo com determinado critério, é atribuído um novo peso a cada partícula do filtro. A Figura 2.5(d) ilustra o reajuste dos pesos das partículas, em que o tamanho da partícula é proporcional ao peso atribuído a ela. No exemplo, o critério utilizado para ajustar o peso foi a proximidade da partícula ao centro de massa real obtido no passo anterior. Por esse motivo, na figura, as partículas mais próximas ao centro real são maiores que as mais distantes.

Por fim, o filtro é corrigido com o intuito de produzir novas partículas (com mesmo peso) mais significantes que as atuais, utilizando os pesos atribuídos às partículas anteriormente como fundamento para essa correção. A Figura 2.5(e) exibe as novas partículas para o exemplo dado. Note como, antes da correção, algumas partículas encontravam-se mais afastadas do último centro de massa observado e, após, essa distância é amenizada. Vale observar que existem vários métodos propostos para lidar com algumas particularidades existentes neste e nos outros estágios do filtro [Mon05].

2.1.4 Extração de Atributos

Extrair as características (atributos) mais importantes numa imagem evidencia as diferenças e similaridades entre os objetos. Entre essas características pode-se incluir o brilho de uma determinada região, a textura de uma região, a amplitude do histograma, entre outros [Pra07].

A extração de atributos visa realizar medidas de atributos dos objetos de interesse que estão na imagem, que são utilizados para distinguir uma classe de padrão. As medidas são adquiridas computacionalmente através de algoritmos que extraem informações de tamanho, cor do objeto, características de forma, conectividade, textura.

Os descritores de formas foram escolhidos neste trabalho por serem métodos matemáticos que representam a forma do objeto ou a região de interesse que estão presentes na imagem. Nesta seção serão discutidos alguns descritores de forma: descritores geométricos, momentos de imagem, k-curvatura.

Descritores Geométricos

Diversos descritores podem ser utilizados para distinção geométrica de uma região ou objeto como: o fator de forma, a circularidade, relação entre os eixos e densidade do objeto. O fator de forma (*form factor*) de um objeto corresponde à divisão da área desse objeto pela área de uma circunferência cujo perímetro corresponde ao perímetro do objeto. A área do objeto pode ser encontrada computacionalmente na imagem. Este atributo é calculado pela Equação 2.3.

$$FF = \frac{A_{obj} \times 4 \times \pi}{P^2} \quad (2.3)$$

A circularidade (*roundness*) mensura o quanto alongado o objeto está. Podemos encontrá-la dividindo a área do objeto pela área de uma circunferência cujo diâmetro é igual ao comprimento do objeto. Para calcular esse comprimento deve-se encontrar uma elipse com o mesmo momento de segunda ordem do objeto. O maior eixo dessa elipse é o comprimento do objeto. Deste modo, a dificuldade ao encontrar a circularidade de um objeto está em encontrar seu comprimento, pois, através dele, encontramos o raio da circunferência, calculamos sua área e, posteriormente, sua circularidade. Esse comprimento também pode ser encontrado de forma computacional [SP05]. Portanto, a circularidade é encontrada com a Equação 2.4.

$$C = \frac{4 \times A_{obj}}{\pi \times comprimento^2} \quad (2.4)$$

Similarmente à circularidade, a relação entre eixos (*aspect ratio*) também mede o alongamento do objeto, porém, por meio de outras medidas do objeto. Para calculá-la, encontram-se os dois eixos da elipse correspondente ao objeto e divide-se o maior pelo menor. Assim como o maior eixo da elipse, o menor eixo também pode ser encontrado. A Equação 2.5 é a relação explicada, em que *largura* corresponde ao menor eixo da elipse, e a Equação 2.6 define a densidade do objeto (*compactness*), em que C corresponde à circularidade desse objeto [Sou10].

$$AR = \frac{\text{comprimento}}{\text{largura}} \quad (2.5)$$

$$D = \sqrt{C} \quad (2.6)$$

Momentos de Imagem

Os momentos de ordem $p + q$ sobre uma função contínua 2D são calculadas pela Equação 2.7.

$$M_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q f(x, y) dx dy \quad (2.7)$$

Pode-se visualizar uma imagem como um função 2D discreta I , em que a intensidade para cada *pixel* é alocada com $I(x, y)$. Na Equação 2.8 é o resultado da derivação da Equação 2.7 aplicada a imagens.

$$M_{pq} = \sum_{x=1}^{\text{width}} \sum_{y=1}^{\text{height}} x^p y^q I(x, y) \quad (2.8)$$

Portanto, a representação de um objeto por momentos deve-se considerar apenas os *pixels* existentes no objeto. Através dos momentos de imagens, aplicados a um objeto, é possível derivar algumas propriedades, como por exemplo, a área do objeto, que é representada pelo momento M_{00} e o centro de massa $\bar{x}$ and $\bar{y}$ representado pelos momentos $\frac{M_{10}}{M_{00}}$, $\frac{M_{01}}{M_{00}}$, respectivamente [SGM⁺].

K-Curvatura

É a representação do contorno de um objeto através da relação do ângulo formado entre dois vetores. A imagem segmentada é percorrida após a aplicação de um detector de bordas. Obtento-se um conjunto de pontos (Equação 2.9).

$$P = \{p_0, p_1, \dots, p_n\} \quad \text{with} \quad p_i = (x_i, y_i) \quad (2.9)$$

A remoção de ruídos do contorno é feita considerando-se todos os pontos extraídos do objeto, sendo selecionados pontos após um espaçamento de k valores. O cálculo do ângulo é feito entre três pontos. O vetor v é formado pelos pontos p_i e p_{i+k} , enquanto o vetor w é formado pelos pontos p_{i+k} e p_{i+2*k} . Na Equação 2.10 apresenta o cálculo do ângulo entre dois vetores [SGM⁺].

$$\theta = \cos^{-1} \frac{v \cdot w}{|v||w|} \quad (2.10)$$

2.1.5 Reconhecimento de Padrões

Embora o reconhecimento de padrões possa ser aplicado a problemas sem qualquer relação com visão computacional, existe uma interseção entre as duas áreas, uma vez que o reconhecimento de padrões é uma importante etapa na solução de problemas de visão computacional [Sou10]. O reconhecimento de padrões é a ciência que estuda a forma de identificação ou classificação de um conjunto de informações [Amo06].

Uma das principais abordagens para reconhecimento de padrões é a aprendizagem supervisionada [Qui09], onde exemplos de cada padrão em um conjunto de informações são fornecidos, ou seja, amostras de casos previamente classificados são utilizados como treinamento [TK99].

Capítulo 3

Desenvolvimento

Durante o desenvolvimento deste trabalho foram implementados dois módulos, o primeiro destinado a realização do rastreamento de múltiplas larvas. Já o segundo módulo recebe a localização das larvas através do módulo anterior e efetua a classificação do estado de movimentação (aberta, curvada ou fechada) da larva.

As implementações dos módulos foram realizadas utilizando a linguagem de programação Java e o aproveitamento de diversos recursos disponíveis na ferramenta ImageJ. O ImageJ é um *software* de código aberto, destinada ao processamento e análise de imagens. Algoritmos de rastreamento e segmentação de imagens foram utilizados da biblioteca SIGUS que é uma plataforma com um conjunto de implementações voltados à área de visão computacional [PMPN06]

3.1 Módulo de Rastreamento de Múltiplas Larvas

Neste módulo foram utilizados os algoritmos de segmentação baseados na subtração de fundo e filtro de partículas, ambos disponíveis na biblioteca SIGUS. É gerado um filtro de partículas para cada larva, usando o mesmo modelo de dinâmica por serem objetos da mesma natureza. O módulo é capaz de retornar valores de relevância para o módulo gerador de sequência de observação como:

- Coordenada X e Y do centro de massa da larva;
- Circularidade da elipse;
- Ângulo de inclinação da elipse;

Um dos maiores problemas durante o rastreamento de múltiplas larvas é a ocorrência de oclusões causada quando as larvas estão muito próximas umas das outras. Um exemplo claro da ocorrência de oclusões pode ser observado nas Figuras 3.1(a) e 3.1(b) onde duas larvas estão se aproximando, já nas Figuras 3.1(c) e 3.1(d) ocorre a oclusão entre as larvas por estarem muito próximas ou sobrepostas. Devido a essa situação que este módulo tem a sua importância.

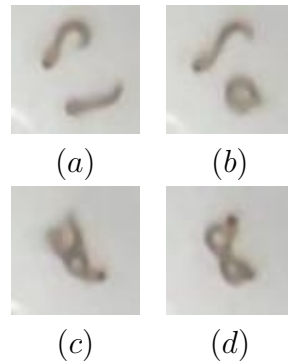


Figura 3.1: Sequência de imagens onde irá ocorrer oclusão

O filtro de partículas irá auxiliar o modelo de observação a identificar corretamente o centro de massa das larvas. Quando a observação assinala que foi encontrado um número menor de regiões do que o previsto, a partícula mais significativa de cada filtro será utilizada para tomar lugar da observações incorretas [Mon05].

3.2 Módulo de Classificação Automática

Para identificar quando uma larva está viva ou morta é necessária a observação do seu movimento. Para isso, o movimento que garante o deslocamento da larva no meio líquido foi decomposto em três possíveis estados. Os estados foram denominados de posição aberta, curvada e fechada.



Figura 3.2: Estados de Movimentação da Larva

Na sequência de imagens da Figura 3.2, a posição aberta Figura 3.2(a) seria o estado inicial da movimentação da larva, a posição curvada Figura 3.2(b) como um estado intermediário e por último a posição fechada Figura 3.2(c) como estado final.

Uma larva é dada como viva quando os estados de movimentação citados anteriormente são realizados rapidamente e com uma elevada frequência de repetições dos estados de movimentação. E quando esses movimentos e a frequência de repetições são reduzidos ou há ausência de movimento durante o tempo de observação então a larva é considerada como morta.

A implementação desse módulo foi dividida em duas etapas. Na primeira etapa, os parâmetros que o módulo de rastreamento retorna são utilizados para localizar a região que está a larva na imagem original onde é realizado um recorte nessa região. Esse recorte então é passado para o extrator de atributos do SIGUS que retorna um vetor de atributos baseado em formas. Esse vetor então é processado por um classificador do WEKA previamente treinado. Tendo assim um dos possíveis estados de movimento da larva (aberta, curvada ou fechada) classificado. A primeira etapa está representada no diagrama da Figura 3.3.

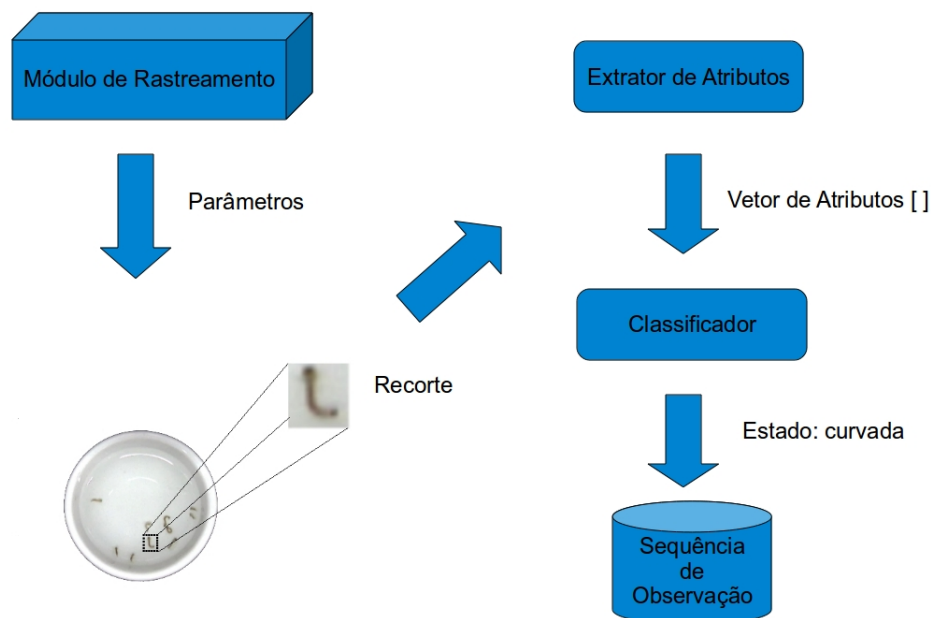


Figura 3.3: Diagrama da primeira etapa.

A segunda etapa ficou responsável por gerar e classificar as sequências de observações. Onde consiste em armazenar uma quantidade determinada de estados de movimentação da larva. Quando essa quantidade determinada é alcançada é gerado um vetor de atributos temporais contendo o número de ocorrências, número de ocorrências sucessivas e número de transições de um estado para outro para cada estado de movimentação da larva.

Então esse novo vetor gerado é submetido a outro classificador também previamente treinado com sequências de observações de larvas vivas e mortas. Estas etapas são realizadas para cada larva identificada pelo módulo de rastreamento.

3.3 Módulo de Estatística

Muitos problemas de inferência quantitativa em pesquisa biológica e tecnológicas dizem respeito à relação entre um estímulo e uma resposta[Gua91]. A estatística torna-se necessária quando a resposta não é determinada exatamente pelo conhecimento do estímulo, e as repetições dos experimentos ou observações para valores estabelecidos da variável independente não apresentam o mesmo tamanho de resposta[Bli38].

A análise probit foi então desenvolvida para dar interpretações de tais resultados. Por esses motivos foi implementado um módulo de estatística para realizar análise probit ao final dos experimentos dos bioensaios. Na Figura 3.4(a) mostra a tela inicial do módulo para entrada de dados, a Figura 3.4(b) o usuário pode visualizar as variáveis calculadas para realizar a análise, na Figura 3.4(c) mostra os resultados da análise probit e na Figura 3.4(d) é gerado um gráfico com a regressão linear dos dados para realizar a análise probit.

3.4 Integração dos Módulos em um Sistema de Informação

Implementado um sistema de informação capaz de integrar os módulos de visão computacional e estatísticas desenvolvidos, para o controle dos diversos experimentos com diferentes extratos que é alimentado pelos resultados do módulo de visão computacional ou pelo próprio usuário. Primeiramente foi implementado um banco de dados utilizando *MYSQL*, para armazenamentos dos dados dos experimentos realizados. No diagrama entidade-relacionamento da Figura 3.5 pode-se observar a estrutura do banco de dados.

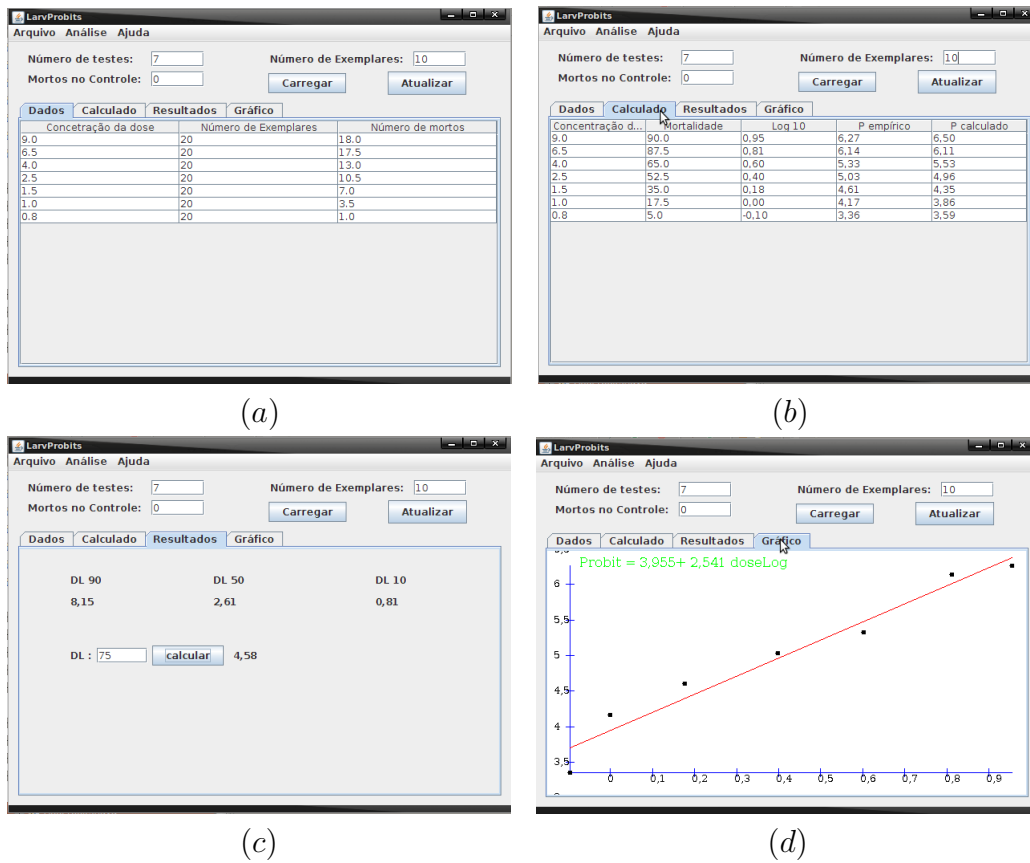


Figura 3.4: Telas do módulo de estatística.

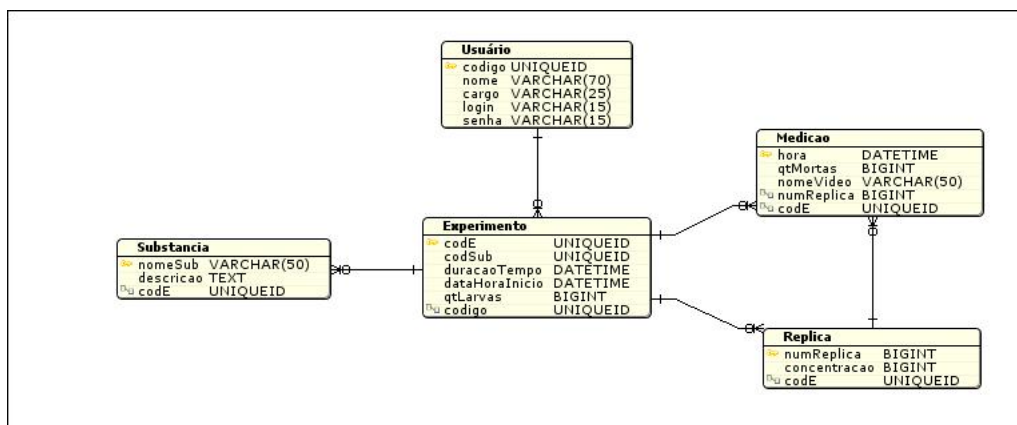


Figura 3.5: Diagrama entidade-relacionamento.

O sistema de informação implementado suporta as operações CRUD(*Create*, *Retrive*, *Update* e *Delete*) para as entidades usuário, experimentos e substâncias. Na Figura3.6 mostra a tela inicial do sistema.



Figura 3.6: Tela inicial.

Capítulo 4

Experimentos

O experimento realizado tem como objetivo testar os extratores de atributos baseados em forma implementados na biblioteca SIGUS por meio dos classificadores do Weka. Assim selecionar os atributos que elevam a taxa de acerto dos classificadores. Foram utilizados os seguintes extratores: Descritores Geométricos, Momentos de Imagem e K-Curvatura.

Foi criada uma base de treinamento constituída por vinte amostras para cada estado de movimentação que foram definidos para a larva. As amostras são recortes individuais de larvas nas imagens de diferentes vídeos onde foram filmadas larvas em meio líquido, exemplos de amostras podem ser vistas nas Figuras 3.2(a), 3.2(b) e 3.2(c). Essas amostras foram submetidas aos extratores mencionados anteriormente. Foram geradas para cada extrator uma base de dados denominada ARFF (*Attribute-Relation File Format*), basicamente esse arquivo contém o conjunto de instâncias e um cabeçalho com os nomes da coleção de instâncias e das características que compõem cada instância.

Também foi gerado um só arquivo ARFF com todos os atributos disponíveis de cada extrator totalizando 48 atributos. Os ARFFs gerados foram processados pelo Weka para o treinamento de seus algoritmos de aprendizagem. Foram testados dois algoritmos, C4.5 (árvore de decisão) e KNN (*k-nearest neighbor*). O Weka possui uma recurso que possibilita realizar testes para medir a taxa de acerto dos classificadores. A Tabela 4.1 exibe a porcentagem atingida individualmente por cada extrator e por último com atributos de todos os extratores.

Outro experimento foi realizado para avaliar os classificadores C4.5 e KNN na classificação do estado vital da larva utilizando os atributos temporais obtidos na sequência de observação. A taxa de acerto foi obtida da seguinte maneira: Um especialista que realiza contagem de larvas vivas e mortas em laboratório assiste um vídeo com 10 larvas entre elas vivas e mortas.

	C4.5	KNN
Momentos de Imagem	71.6	76.6
K-Curvatura	65.0	73.3
Descritores Geométricos	88.0	90.0
Todos	71.0	65.0

Tabela 4.1: Taxa de acerto dos extratores.

Inicialmente foi adotado que todas as larvas estivessem vivas, no decorrer do vídeo o especialista marca o trecho do vídeo em que foi notado a mudança do estado de viva para morta e o número de larvas mortas.

Por exemplo, o especialista está assistindo o vídeo quando no quadro 500 ele percebe que uma larva está morta então é marcado quadro 500 e larvas mortas 1, passado um certo tempo é observado que no quadro 2760 mais uma larva e considerada morta, então é marcado quadro 2760 e larvas mortas 2. Com esse exemplo podemos dizer que do início do vídeo até o quadro 499 todas as larvas estão vivas, dos quadros 500 até 2759 apenas uma morta e após do 2760 adiante duas mortas, tendo assim os intervalos em que podemos afirmar que há uma número x de larvas vivas e um número y de mortas.

Então foram extraídas 10 sequências de observações com tamanhos de 120, 240, 300, 360 e 420 quadros desses intervalos. Para calcular a taxa de acerto seria então a razão entre a taxa de mortalidade classificada com o sistema pela taxa de mortalidade marcada pelo especialista do intervalo em que foi retirada a sequência de observação. Onde a taxa de mortalidade é a razão entre número de larvas mortas pelo número total de larvas, no caso o vídeo havia 10 larvas no recipiente.

Na Tabela 4.2 mostra a média da taxa de acerto de 10 sequências de observações com tamanhos distintos obtida pelos classificadores. Este experimento mostra que dada uma sequência de observações de maior tamanho os atributos temporais gerados tem melhores ajustes quando submetidos ao classificador.

	C4.5	KNN
120	27.5	30
240	40	42.5
300	45	50
360	65	77.5
420	85	92.5

Tabela 4.2: Média da taxa acerto para cada 10 sequências de observações de tamanhos diferentes.

Capítulo 5

Considerações Finais

Este trabalho mostrou os métodos necessários para a implementação de um sistema de visão computacional para se automatizar uma tarefa que está bastante sujeita a erros por ser cansativa para o ser humano. Foram abordadas técnicas de visão computacional como subtração de fundo, filtro de partícula e extração de atributos baseado em forma que foram utilizadas durante o desenvolvimento do sistema.

Módulos para rastreamento e classificação automática foram implementados, onde são responsáveis pela produção dos principais parâmetros para o processo que classifica larvas mortas e vivas durante os teste com os larvicidas. Esses módulos de visão computacional foram integrados a um sistema de informação juntamente com um módulo de estatística, para o controle dos diversos experimentos com diferentes extratos.

Referências Bibliográficas

- [Amo06] W. P. Amorim. Extração de atributos e análise discriminante aplicadas na detecção de defeitos em couro bovino. Monografia (Graduação), 2006. Universidade Católica Dom Bosco, Campo Grande, MS.
- [Bli38] C.I. Bliss. The determination of the dosage-mortality curve from small numbers. *Quarterly Journal of Pharmacology*, 1938.
- [eNVH99] Marques F. O. e Neto V. H. *Processamento Digital de Imagens*. 1 edition, 1999.
- [GOL04] S. K. GOLDENSTEIN. A gentle introduction of predictive filters. *Revista da Informática Teórica e Aplicada (RITA)*, 2004.
- [Gua91] José Roberto Guadanhin. O método probit aplicado a ensaios biológicos. Master's thesis, Universidade de São Paulo, 1991.
- [GW00] R. C. Gonzalez and R. E. Woods. *Processamento de Imagens Digitais*. Primeira edition, 2000.
- [GWE06] R. C. Gonzalez, R. E. Woods, and S. L Eddins. *Digital Image Processing using MATLAB*. 2006.
- [HLCP00] C. Hue, J.P. Le Cadre, and P. Perez. Tracking multiple objects with particle filtering. 2000.
- [inf09] Secretaria de vigilância em saúde informe epidemiológico da dengue semanas de 1 a 52 de 2009. Technical report, 2009.
- [Mon05] J. B. O. Monteiro. Filtro de partículas para rastreamento de múltiplos camundongos. Monografia (Graduação), 2005. Universidade Católica Dom Bosco, Campo Grande, MS.

- [PMPN06] H. Pistori, P. S. Martins, M. C. Pereira, and J. J. Neto. Sigus - plataforma de apoio ao desenvolvimento de sistemas para inclusão digital de pessoas com necessidades especiais. In *IV Congresso Iberdiscap: Tecnologias de Apoio a Portadores de Deficiência*, Vitória, ES, February 2006.
- [Pra07] William K. Pratt. *Digital Image Processing*. Fourth edition, 2007.
- [Qui09] L. N. B. Quinta. Desenvolvimento de um sistema de visão computacional para o controle microbiano em processos de produção de etanol. Monografia (Graduação), 2009. Universidade Católica Dom Bosco, Campo Grande, MS.
- [Rek04] Ioannis M. Rekleitis. A particle filter tutorial for mobile robot localization. Technical Report TR-CIM-04-02, Centre for Intelligent Machines, McGill University, 3480 University St., Montreal, Québec, CANADA H3A 2A7, 2004.
- [SGM⁺] J. A. Silva, W. N. Gonçalves, B. B. Machado, H. Pistori, and A. S. Souza. Comparação de descritores de formas no reconhecimento de objetos. *III WVC - Workshop de Visão Computacional, São José do Rio Preto, São Paulo*.
- [Sou10] Kleber Padovani Souza. Aplicação de modelos de markov ocultos na obtenção de taxas de mortalidade das larvas do mosquito da dengue. Master's thesis, Universidade Federal de Mato Grosso do Sul, 2010.
- [SP05] K. P. Souza and H. Pistori. Implementação de um extrator de características baseado em momentos da imagem. *SIBGRAPI '05: Proceedings of the 18th Brazilian Symposium on Computer Graphics and Image Processing*, 2005.
- [TK99] Sergios Theodoridis and Konstantinos Koutroumbas. *Pattern recognition*. San Diego : Academic press, 1999., 1999.
- [ÉST09] Érika Sayuri Tezuka. Um modelo de visão computacional para identificação do estágio de maturação e injúrias no pós-colheita de bananas (*musa sapientum*). Master's thesis, Universidade Federal de São Carlos, 2009.