



**UNIVERSIDADE FEDERAL DE PERNAMBUCO**  
**CENTRO DE INFORMÁTICA**  
**CURSO DE BACHARELADO EM ENGENHARIA DA COMPUTAÇÃO**

**DIÓGENES DOS SANTOS SILVA**

**AJUSTES DO ELEMENTO ESTRUTURAL E FILTROS PARA  
CONTAGEM AUTOMÁTICA DE OVOS DO AEDES AEGYPTI EM  
IMAGENS DE ARMADILHAS DE OVOS**

**RECIFE**

**2017**

**UNIVERSIDADE FEDERAL DE PERNAMBUCO**  
**CENTRO DE INFORMÁTICA**  
**CURSO DE BACHARELADO EM ENGENHARIA DA COMPUTAÇÃO**

**DIÓGENES DOS SANTOS SILVA**

**AJUSTES DO ELEMENTO ESTRUTURAL E FILTROS PARA  
CONTAGEM AUTOMÁTICA DE OVOS DO AEADES AEGYPTI EM  
IMAGENS DE ARMADILHA DE OVOS**

Monografia apresentada ao Centro de Informática (CIN) da Universidade Federal de Pernambuco (UFPE), como requisito parcial para conclusão do Curso de Engenharia da Computação, orientada pelo professor Leandro Maciel Almeida.

**RECIFE**

**2017**

**UNIVERSIDADE FEDERAL DE PERNAMBUCO**  
**CENTRO DE INFORMÁTICA**  
**CURSO DE BACHARELADO EM ENGENHARIA DA COMPUTAÇÃO**

**DIÓGENES DOS SANTOS SILVA**

**AJUSTES DO ELEMENTO ESTRUTURAL E FILTROS PARA  
CONTAGEM AUTOMÁTICA DE OVOS DO Aedes Aegypti EM  
IMAGENS DE ARMADILHAS DE OVOS**

Monografia submetida ao corpo docente da Universidade Federal de Pernambuco.

Banca Examinadora:

---

Leandro Maciel Almeida

Doutor

Orientador

---

Cleber Zanchettin

Doutor

Examinador

“A maioria dos homens prefere negar uma  
verdade dura do que enfrentá-la”  
George R. R. Martin

## RESUMO

O *Aedes Aegypti* é o mosquito responsável pela transmissão dos vírus da Zika, Chikungunya e Dengue, sendo este último o responsável por mais de 1.500.000 casos em todo o Brasil no ano de 2016. Devido à falta de uma vacina e tratamentos efetivos, a única forma de reduzir a proliferação do vírus é a redução dos criadouros do mosquito. Um dos métodos mais eficientes para a detecção e vigilância de mosquitos são as armadilhas chamadas de *ovitampas*. A contagem dos ovos dos mosquitos nessas armadilhas é extremamente importante, porém essa contagem ainda é feita de forma não-automática. Este trabalho tem como objetivos:

- Fazer um estudo comparativo entre os métodos de contagem propostos na literatura e os métodos de contagem que foram desenvolvidos neste trabalho para um conjunto de imagens de *ovitampas* capturadas através de uma câmera digital;
- Verificar a influência da inclinação do elemento estruturante utilizado nas operações morfológicas, etapa importante no processo de contagem automática;
- Avaliar o desempenho na contagem de ovos em divisões das imagens de *ovitampas*;
- Aplicar os algoritmos de contagem deste trabalho, desenvolvidos para as imagens da câmera digital, em um novo conjunto de imagens obtidos através de um microscópio digital.

**Palavras-chave:** *Aedes Aegypti*. Processamento digital de imagens. Segmentação. Binarização. *Ovitampas*.

## Sumário

1.	Introdução.....	9
1.1.	Contexto e motivação .....	9
1.2.	Objetivos .....	10
1.3.	Estrutura do trabalho .....	11
2.	Conceitos Básicos .....	12
2.1.	Imagem Digital .....	12
2.2.	Modelo de Cores .....	12
2.2.1	Modelo de Cores RGB .....	13
2.2.2	Modelo de Cores YIQ .....	13
2.2.3	Modelo de Cores HSV .....	14
2.2.4	Modelo de Cores CIE L*a*b.....	15
2.3.	Segmentação de Imagens e Binarização .....	15
2.4.	Métodos de Binarização.....	16
2.5	Operações Morfológicas .....	16
3	Materiais e Métodos .....	21
3.1	Aquisição das Imagens .....	21
3.2	Método 1.....	22
3.3	Método 2.....	26
3.4	Método 3.....	28
3.5	Contagem de Ovos para Diferentes Inclinações do Elemento Estruturante .....	30
3.6	Contagem de Ovos em Divisões de Imagens de Ovitampas.....	30
3.7	Contagem de Ovos para um Novo Conjunto de Imagens .....	31
3.7.1	Contagem para Imagens com Ampliação.....	32
3.7.2	Contagem para Imagens sem Ampliação.....	33
4	Resultados Experimentais .....	35
4.1	Experimento 1 .....	35
4.2	Experimento 2 .....	36
4.3	Experimento 3 .....	39
4.4	Experimento 4 .....	40
5	Conclusões e Trabalhos Futuros.....	44
5.1	Contribuições .....	44
5.2	Trabalhos Futuros.....	45
	Bibliografia .....	46
	Apêndice A .....	48
	Apêndice B .....	52

## Lista de Figuras

Figura 1 - Exemplo de uma Ovitrapa.....	10
Figura 2 - Modelo de cores RGB .....	13
Figura 3 - Modelo de cores YIQ .....	14
Figura 4 - Modelo de cores HSV .....	14
Figura 5 - Modelo de cores $L^*a^*b$ .....	15
Figura 6 - Exemplo da operação de binarização .....	16
Figura 7 - Exemplo dos conceitos de reflexão (centro) e translação (direita).....	17
Figura 8 - Exemplos de elementos estruturantes.....	18
Figura 9 – Exemplo da operação de erosão.....	18
Figura 10 – Exemplo da operação de dilatação.....	19
Figura 11 - Exemplo da operação de abertura.....	19
Figura 12 - Exemplo da operação de fechamento .....	20
Figura 13 - Imagem utilizada como exemplo para os capítulos 3.2 ao 3.6 .....	22
Figura 14 - Imagem com ampliação (esquerda) e imagem sem ampliação (direita) utilizadas como exemplo para o capítulo 3.7 .....	22
Figura 15 - Componentes do modelo HSV para a imagem da Figura 3: tonalidade (topo-esquerda), saturação (topo-direita) e valor (abaixo) .....	23
Figura 16 - Componente da tonalidade após a binarização com o método de Otsu (esquerda) e Kittler (direita).....	24
Figura 17 - Eliminação das pequenas áreas da imagem binarizada pelo método de Otsu (esquerda) e Kittler (direita) .....	24
Figura 18 - Ovo médio (esquerda) utilizado para a definição do elemento estruturante (direita).....	25
Figura 19 - Imagem resultante da operação morfológica para as imagens binarizadas pelo método de Otsu (esquerda) e Kittler (direita) .....	25
Figura 20 - Fluxo de execução do método 1 .....	26
Figura 21 - Componente I do modelo YIQ (topo-esquerda), imagem alterada para fácil visualização das diferentes áreas (topo-direita) e a binarização com threshold fixo (abaixo-esquerda) .....	27
Figura 22 - Remoção das áreas inferiores a 170 pixels e aplicação da operação morfológica de fechamento .....	28
Figura 23 - Fluxo de execução do método 2 .....	28
Figura 24 - Componente da luminosidade (L) do modelo $L^*a^*b$ (topo-esquerda), imagem alterada para fácil visualização das diferentes áreas (topo-direita) e a binarização com threshold fixo (abaixo) .....	29
Figura 25 - Fluxo de execução para o método 3 .....	29
Figura 26 - Exemplo de uma ovitrapa completa .....	30
Figura 27 – Processo de divisão da Figura 3 como uma matriz quadrada de ordem 3 .....	31
Figura 28 - Componente I do modelo YIQ (topo-esquerda) da imagem da Figura 4 (esquerda), binarização com threshold fixo (topo-direita), remoção de pequenas áreas (baixo-esquerda) e aplicação da função imdelate (baixo-direita) .....	33
Figura 29 – Componente I do modelo YIQ (topo-esquerda) da imagem da Figura 4 (direita), binarização com threshold fixo (topo-direita), remoção de pequenas áreas (baixo-esquerda) e aplicação da função imdelate (baixo-direita) .....	34
Figura 30 - Imagem da ovitrapa contendo 111 ovos.....	39
Figura 31 - Imagem que gerou discrepância no resultado do método 1 com Kittler .....	43
Figura 32 - Imagem após ajustes de luminosidade e aplicação da binarização de Kittler .....	43

## LISTA DE TABELAS

Tabela 1 – Resultado da contagem para os métodos desenvolvidos dos capítulos 3.2 ao 3.4 .....	35
Tabela 2 - Resultado da contagem rotacionando o elemento estruturante para uma imagem com 8 OVOS .....	37
Tabela 3 - Resultado da contagem rotacionando o elemento estruturante para uma imagem com 19 OVOS .....	37
Tabela 4 - Resultado da contagem rotacionando o elemento estruturante para uma imagem com 22 OVOS .....	37
Tabela 5 - Resultado da contagem rotacionando o elemento estruturante para uma imagem com 34 OVOS .....	38
Tabela 6 - Resultado da contagem rotacionando o elemento estruturante para uma imagem com 111 OVOS .....	38
Tabela 7 - Resultado das divisões da imagem da Figura 3.....	39
Tabela 8 - Resultado das divisões da imagem da Figura 16.....	40
Tabela 9 - Resultado da contagem para os métodos desenvolvidos dos capítulos 3.7 para imagens com ampliação .....	42
Tabela 10 - Resultado da contagem para os métodos desenvolvidos dos capítulos 3.7 para imagens sem ampliação.....	42



## 1. Introdução

Este capítulo foi dividido em três partes: inicialmente será apresentado o contexto em que se inclui o trabalho e qual sua motivação, em seguida os objetivos almejados e finalmente uma breve descrição da estrutura do documento.

### 1.1. Contexto e motivação

O *Aedes Aegypti* é um mosquito originário do continente africano, mais precisamente do Egito, e teve sua disseminação para as regiões tropicais e subtropicais em meados do século XVI devido as grandes embarcações que levavam escravos para estas regiões. Os primeiros relatos do mosquito no Brasil datam do final do século XIX e início do século XX.

O *Aedes Aegypti* é o responsável pela transmissão dos vírus da *Zika*, *Chikungunya* e Dengue, sendo este último o responsável por mais de 1.500.000 casos em todo o território nacional no ano de 2016 segundo os dados do ministério da saúde [1]. O vírus da dengue pode se manifestar em seres humanos de quatro maneiras distintas, dentre essas a febre hemorrágica é o tipo mais grave e pode levar o indivíduo à morte.

Cada mosquito pode viver em média 30 dias e as fêmeas chegam a colocar entre 130 a 200 ovos. Os ovos necessitam de água para eclodirem, portanto, as fêmeas optam por pôr seus ovos em recipientes que possuam a capacidade de reter água, como por exemplo garrafas vazias, pneus e caixas d'água descobertos.

Na falta de uma vacina e tratamentos efetivos, a única forma de frear a proliferação do vírus é a redução dos criadouros do mosquito. Dentre os vários métodos existentes de detecção e monitoramento, o uso de armadilha de ovos (*ovitrampas*) tem sido um dos mais eficientes devido ao seu baixo custo e sua alta capacidade de detecção da presença de mosquitos mesmo em situação de baixa infestação [2]. As armadilhas consistem em recipientes pretos preenchido parcialmente com água e uma palheta áspera de madeira instalada verticalmente em seu interior (vide Figura 1).

Figura 1 - Exemplo de uma *Ovitrapa*



*Fonte:* <http://www.fiocruz.br/ioc/cgi/cgilua.exe/sys/start.htm?infoid=2135&sid=32&tpl=printerview>

A contagem de ovos nestas armadilhas fornece dados importantes para que agências do governo e programas de controle de doenças desenvolvam planos de ações para o controle da expansão de casos da dengue. Esta contagem é normalmente feita de forma manual, visual e não-automática [3].

## 1.2. Objetivos

Com o intuito de tornar essa contagem automática, e conseqüentemente auxiliar no controle da propagação de casos da dengue, métodos foram propostos por [3] utilizando técnicas de processamento de imagens em um conjunto de imagens de *ovitrampas*. Com o propósito de melhorar os resultados obtidos, este trabalho tem como objetivos:

- Fazer um estudo comparativo entre os métodos de contagem propostos na literatura e os métodos de contagem que foram desenvolvidos neste trabalho para um conjunto de imagens de *ovitrampas* capturadas através de uma câmera digital;
- Verificar a influência da inclinação do elemento estruturante utilizado nas operações morfológicas, etapa importante no processo de contagem automática;
- Avaliar o desempenho na contagem de ovos em divisões das imagens de *ovitrampas*;
- Aplicar os algoritmos de contagem deste trabalho, desenvolvidos para as imagens da câmera digital, em um novo conjunto de imagens obtidos através de um microscópio digital.

### **1.3. Estrutura do trabalho**

Este trabalho está dividido em cinco capítulos e dois apêndices:

Capítulo 1 – Introdução – este capítulo apresenta o contexto que se inclui o trabalho e os objetivos almejados.

Capítulo 2 – Conceitos Básicos – este capítulo apresenta termos e conceitos utilizados em processamento de imagens para a fácil compreensão do presente trabalho.

Capítulo 3 – Materiais e Métodos – este capítulo mostra como as imagens utilizadas neste trabalho foram obtidas e como foram desenvolvidos os métodos para alcançar os objetivos.

Capítulo 4 – Resultados Experimentais – este capítulo apresenta os resultados e avaliações dos experimentos realizados sobre os métodos desenvolvidos no capítulo 3.

Capítulo 5 – Conclusões e Trabalhos Futuros – este capítulo apresenta as considerações finais, limitações encontradas e trabalhos futuros.

Apêndice A – este apêndice apresenta as imagens utilizadas nos métodos de contagem.

Apêndice B – este apêndice apresenta os algoritmos implementados no MATLAB® [4].

## 2. Conceitos Básicos

Neste capítulo são introduzidos alguns termos e conceitos utilizados em processamento de imagens com o objetivo de facilitar o entendimento do presente trabalho.

### 2.1. Imagem Digital

Imagens são representações visuais de pessoas ou objetos, estas representações podem ser realizadas de algumas maneiras, como por exemplo pela fotografia. A imagem digital nada mais é que a representação destas imagens do mundo real em um formato digital onde um computador possa armazená-la/processá-la. Para tornar uma imagem em digital, conceitos como quantização e amostragem são necessários. A amostragem define o tamanho da matriz que representará a imagem, enquanto que, a quantização informa a quantidade de cores distintas que a imagem possuirá. Simplificando, temos que a amostragem converte a imagem real em uma matriz  $M \times N$  conforme a seguir:

$$f = \begin{bmatrix} f(1,1) & f(1,2) & \dots & f(1,m) \\ f(2,1) & f(2,2) & \dots & f(2,m) \\ \vdots & \vdots & \vdots & \vdots \\ f(n,1) & f(n,2) & \dots & f(n,m) \end{bmatrix}$$

Onde cada um dos elementos desta matriz é chamado de *pixel* e quanto maior for o tamanho desta matriz maior resolução ela terá, ou seja, a imagem digital comportará um maior nível de detalhes. Finalmente, o processo de quantização faz com que cada um dos *pixels* assumam valores inteiros informando a quantidade de cores que a imagem possuirá.

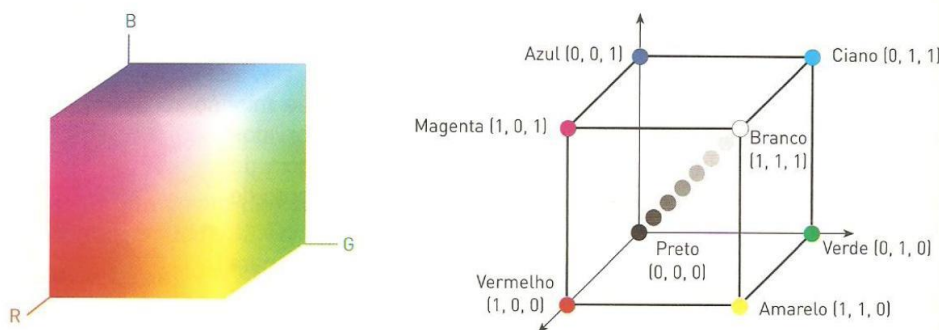
### 2.2. Modelo de Cores

De acordo com [5] o objetivo dos modelos de cores é permitir a especificação de cores em um formato padronizado e aceito por todos. Em linhas gerais, um modelo de cores é uma representação tridimensional na qual cada cor é representada por um ponto do sistema de coordenadas 3-D. Ainda segundo [5] a grande parte dos modelos tem sua utilização dirigida para *hardware* (exs: impressoras e monitores coloridos) ou *softwares* que utilizam manipulações de cores. Por serem padrões, estes modelos podem ser facilmente convertidos de uns para outros, permitindo a flexibilidade em sua utilização. A seguir serão apresentados alguns modelos de cores que foram utilizados neste trabalho:

### 2.2.1 Modelo de Cores RGB

Conforme [5] o modelo RGB (do inglês *Red, Green, Blue* – Vermelho, Verde, Azul) é baseado em um sistema de coordenadas cartesianas, que pode ser visto como um cubo onde três de seus vértices são as cores primárias, outros três as cores secundárias, o vértice junto à origem é o preto e o mais afastado da origem corresponde à cor branca, conforme ilustrado na Figura 2. O modelo RGB é o mais utilizado por câmeras e monitores de vídeo [5].

Figura 2 - Modelo de cores RGB

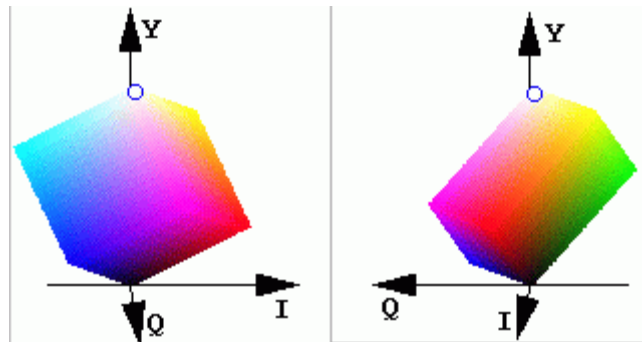


Fonte: <http://asaplicacoesinformaticas.blogspot.com.br/2011/02/modelo-rgb-e-suas-aplicacoes.html>

### 2.2.2 Modelo de Cores YIQ

O modelo YIQ é utilizado no padrão NTSC de TV em cores [5]. Este modelo foi desenvolvido para garantir que os televisores em preto e branco, conseguissem receber o mesmo sinal de televisão que os televisores em cores. Ele é representado como um cubo onde seus vértices representam as cores primárias e secundárias e sua diagonal principal a escala de cinza, conforme ilustrado na Figura 3. O componente Y (luminância) contém a informação do brilho, enquanto as componentes I e Q codificam as informações de cromaticidade [5] que é o grau de pureza da cor.

Figura 3 - Modelo de cores YIQ

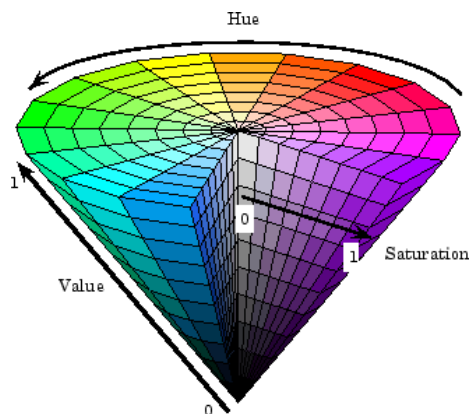


Fonte: <http://asaplicacoesinformaticas.blogspot.com.br/2011/02/modelo-rgb-e-suas-aplicacoes.html>

### 2.2.3 Modelo de Cores HSV

O modelo HSV (do inglês *Hue*, *Saturation*, *Value* – Tonalidade, Saturação, Valor) toma como base o modelo de percepção de cor pelo ser humano [5]. Ele é normalmente representado como um cone com sua base para cima, onde: o perímetro da base representa os valores da tonalidade, o raio representa os valores da saturação e a altura os valores do valor, como pode ser visto na Figura 4. O componente da tonalidade o espectro de cor da imagem, a saturação (também chamada de pureza) define os níveis de tom de cinza que ela terá e finalmente, o valor define o brilho da cor.

Figura 4 - Modelo de cores HSV

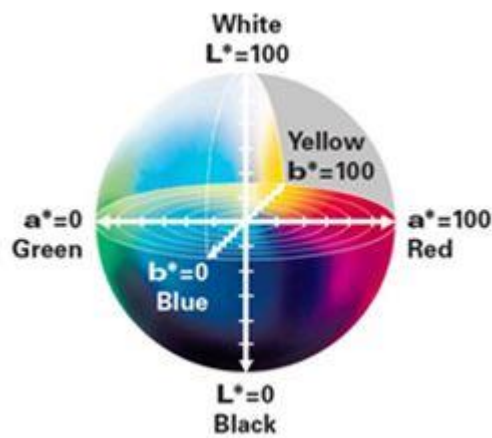


Fonte: <https://www.mathworks.com/help/images/convert-from-hsv-to-rgb-color-space.html?requestedDomain=www.mathworks.com>

## 2.2.4 Modelo de Cores CIE L\*a\*b

Leão [6] cita que o modelo de cores CIE L\*a\*b foi criado pela CIE (*Commission Internationale de l'éclairage*) para aumentar a uniformidade das cores percebidas pelo sistema visual humano. O componente L\* representa a luminosidade da cor, enquanto que os componentes a\* e b\* indicam a informação da cromaticidade [8]. Ele é normalmente representado como uma esfera onde como podemos ver na imagem a seguir:

Figura 5 - Modelo de cores L\*a\*b



*Fonte: <https://www.perten.com/Publications/Articles/Using-the-DA-7300-Digital-Camera-for-Color-Detection-in-Agri-Industry-Processing/>*

## 2.3. Segmentação de Imagens e Binarização

De acordo com [5] a tarefa básica da segmentação é a de dividir uma imagem em suas unidades significativas, ou seja, a segmentação divide a imagem em partes ou objetos para posteriores extrações de medidas ou informações. A binarização, também conhecido como limiarização ou *thresholding*, é a operação mais comum para o processo de segmentação e consiste em separar as regiões de uma imagem quando esta apresenta duas classes (o fundo e o objeto)[5].

A ideia básica do método é definir um valor de limiar, onde cada pixel receberá o valor 1 caso tenha valor maior que o valor de limiar, caso contrário, lhe será atribuído o valor 0. A imagem a seguir exemplifica o processo de binarização:

Figura 6 - Exemplo da operação de binarização



Fonte: <https://maalencar.wordpress.com/2012/06/14/image-segmentation/>

## 2.4. Métodos de Binarização

Os métodos de binarização são divididos em duas categorias: global e local. Os métodos globais definem um limiar para toda a imagem, enquanto que os métodos locais dividem a imagem em áreas e um limiar é definido para cada região ou *pixel*. Os métodos locais possuem uma grande capacidade de adaptação e apresentam os melhores resultados, porém demandam mais tempo e capacidade de processamento. Este trabalho utilizará os métodos globais, pois estes necessitam menos recursos de processamento do que na abordagem local. Os métodos globais *Otsu* [7][12], *Kittler e Illingworth* [7][13] e do *Threshold Fixo* [7] foram utilizados no processo de segmentação dos experimentos deste trabalho.

De maneira geral, os métodos de *Otsu* e *Kittler* definem automaticamente um valor de limiar para a imagem baseada nos valores de cores da imagem, enquanto que o *Threshold Fixo* o valor de limiar deve ser definido manualmente.

## 2.5 Operações Morfológicas

De acordo com [14] a morfologia matemática é uma ferramenta utilizada para extrair componentes das imagens que são úteis na representação e na descrição da forma de uma região. A morfologia toma como base a teoria dos conjuntos para representar os objetos em uma imagem e extrair informações. Os conjuntos em morfologia matemática representam os objetos encontrados em uma imagem, como por exemplo, os pixels brancos em uma imagem binária [14].



A morfologia digital possui duas operações fundamentais: a dilatação e a erosão. Porém, para compreender estas operações três conceitos são necessários: a reflexão, a translação e o elemento estruturante.

A reflexão de um conjunto  $B$ , indicada por  $(B)_r$ , é definida como [14]:

$$(B)_r = \{ w \mid w = -b, b \in B \}$$

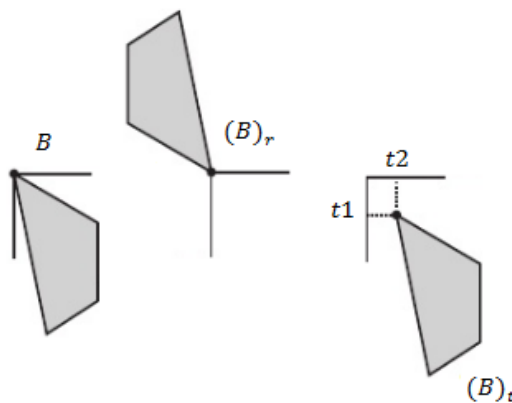
Ou seja, se  $B$  é o conjunto de pixels que representa um objeto em uma imagem, então  $(B)_r$  é simplesmente o conjunto de pontos em  $B$  cujas coordenadas  $(x, y)$  foram substituídas por  $(-x, -y)$  [14].

A translação de um conjunto de pixels  $B$  (que representa um objeto em uma imagem) no ponto  $t = (t1, t2)$ , indicada por  $(B)_t$  é definida como [14]:

$$(B)_t = \{ c \mid c = b + t, \text{ para } b \in B \}$$

Ou seja,  $(B)_t$  é o conjunto de pontos em  $B$ , cujas coordenadas  $(x, y)$  foram substituídas por  $(x + t1, y + t2)$  [14]. A imagem a seguir ilustra os conceitos de reflexão e translação.

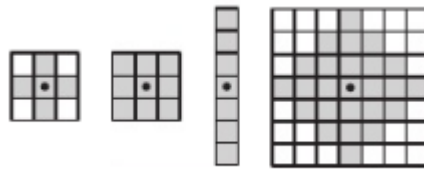
Figura 7 - Exemplo dos conceitos de reflexão (centro) e translação (direita)



**Fonte:** R.C. Gonzalez, R.E. Woods, "Processamento digital de imagens", 3. ed. São Paulo: Pearson, p. 644, 2010

A translação e a reflexão são empregadas nas operações morfológicas tomando como base os chamados elementos estruturantes. Estes elementos são pequenos conjuntos ou subimagens usadas para examinar uma imagem buscando propriedades de interesse [14]. A figura a seguir mostra alguns exemplos de elementos estruturantes.

Figura 8 - Exemplos de elementos estruturantes



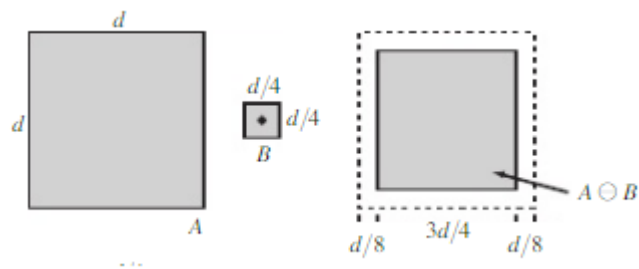
**Fonte:** R.C. Gonzalez, R.E. Woods, "Processamento digital de imagens", 3. ed. São Paulo: Pearson, p. 644, 2010

Tendo em mãos estes conceitos, vamos falar sobre a erosão e a dilatação. A erosão de um conjunto A por B, indicada por  $A \ominus B$  é definida como [14]:

$$A \ominus B = \{ t \mid (B)_t \subseteq A \}$$

Em outras palavras, a erosão de A pelo elemento estruturante B é o conjunto de todos os pontos de t de forma que B, transladado por t, esteja contido em A [14]. A imagem a seguir exemplifica a operação de erosão.

Figura 9 – Exemplo da operação de erosão



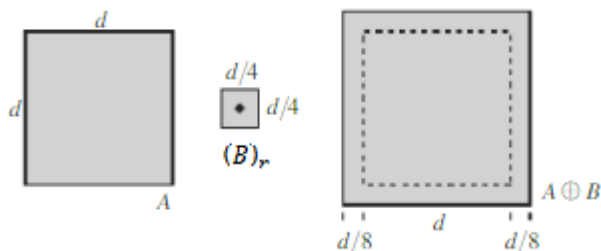
**Fonte:** R.C. Gonzalez, R.E. Woods, "Processamento digital de imagens", 3. ed. São Paulo: Pearson, p. 644, 2010

Já a dilatação de um conjunto A por B, indicada por  $A \oplus B$  é definida como [14]:

$$A \oplus B = \{ t \mid [(B)_r]_t \cap A \neq \emptyset \}$$

Ou seja, a dilatação de A pelo elemento estruturante B é o conjunto de todos os deslocamentos, t, de forma que  $(B)_r$  e A se sobreponham pelo menos por um elemento [14]. A imagem a seguir exemplifica esta operação.

Figura 10 – Exemplo da operação de dilatação



**Fonte:** R.C. Gonzalez, R.E. Woods, “Processamento digital de imagens”, 3. ed. São Paulo: Pearson, p. 644, 2010

A partir destas duas operações outras operações podem ser definidas, como as operações de abertura e fechamento. A abertura, em geral, suaviza o contorno de uma imagem, quebra istmos estreitos e elimina proeminências delgadas [5], enquanto que o fechamento funde pequenas quebras, alarga golfos estreitos, elimina pequenos orifícios e preenche *gaps* no contorno [5].

A abertura de um conjunto A pelo elemento estruturante B, indicado por  $A \circ B$  é definida como [14]:

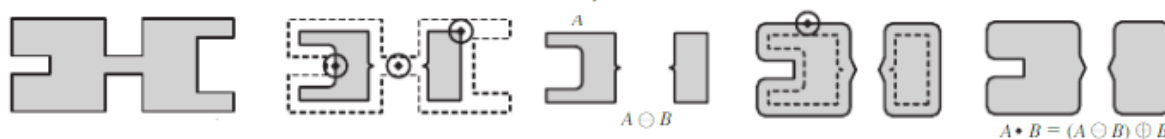
$$A \circ B = (A \ominus B) \oplus B$$

Assim, a abertura de A por B é a erosão de A por B, seguida de uma dilatação do resultado por B [14]. Da mesma forma, o fechamento do conjunto A pelo elemento estruturante B, indicado por  $A \cdot B$  é definido como [14]:

$$A \cdot B = (A \oplus B) \ominus B$$

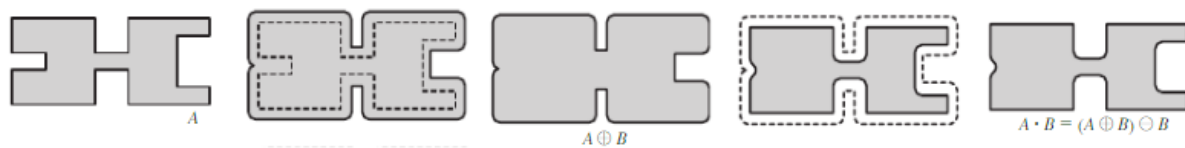
Ou seja, o fechamento de A por B é simplesmente a dilatação de A por B, seguida pela erosão do resultado por B [14]. As imagens a seguir demonstram as operações de abertura e fechamento (respectivamente) onde o elemento estruturante é um pequeno círculo.

Figura 11 - Exemplo da operação de abertura



**Fonte:** R.C. Gonzalez, R.E. Woods, “Processamento digital de imagens”, 3. ed. São Paulo: Pearson, p. 644, 2010

Figura 12 - Exemplo da operação de fechamento



**Fonte:** R.C. Gonzalez, R.E. Woods, "Processamento digital de imagens", 3. ed. São Paulo: Pearson, p. 644, 2010

### 3 Materiais e Métodos

Este capítulo mostra como foram obtidas as imagens das *ovitrampas* utilizadas nos experimentos e também como foram realizados os métodos para alcançar os objetivos definidos.

#### 3.1 Aquisição das Imagens

Para a realização dos experimentos, dois conjuntos de imagens foram utilizados. O primeiro, oriundo dos experimentos realizados por [3], foi obtido através de uma câmera digital com resolução de 7.2 megapixels e zoom óptico de 4.5 vezes. A *ovitampa* foi digitalizada com uma resolução de aproximadamente 700 dpi e com 4 vezes o zoom óptico. Ao final deste processo a imagem digital resultante, possuindo um tamanho de 3,072 por 2,304 pixels, foi recortada gerando um conjunto de 7 imagens (vide Apêndice A - Anexo 1) utilizando o modelo de cores RGB.

O segundo conjunto de imagens foi adquirido através de um microscópio digital com resolução de 2 megapixels, zoom digital de até 6 vezes e uma gama de ampliação de até 1000 vezes. As imagens foram capturadas percorrendo-se a *ovitampa* com o microscópio a uma distância de aproximadamente 0.5 cm e digitalizadas a partir de um software proprietário incluso no conjunto de acessórios do equipamento. Ao final de todo o processo de captura, o segundo conjunto possuía um total de 19 imagens (vide Apêndice A - Anexo 2) e todas estas utilizaram o modelo de cores RGB.

Todos os métodos que serão apresentados a seguir foram desenvolvidos no programa MATLAB® [4] e podem ser vistos no Apêndice B. Com o objetivo de facilitar a compreensão destes métodos, a Figura 13 será utilizada como exemplo nos métodos dos capítulos 3.2 ao 3.6, enquanto que as imagens da Figura 14 serão utilizadas no capítulo 3.7.



Figura 13 - Imagem utilizada como exemplo para os capítulos 3.2 ao 3.6

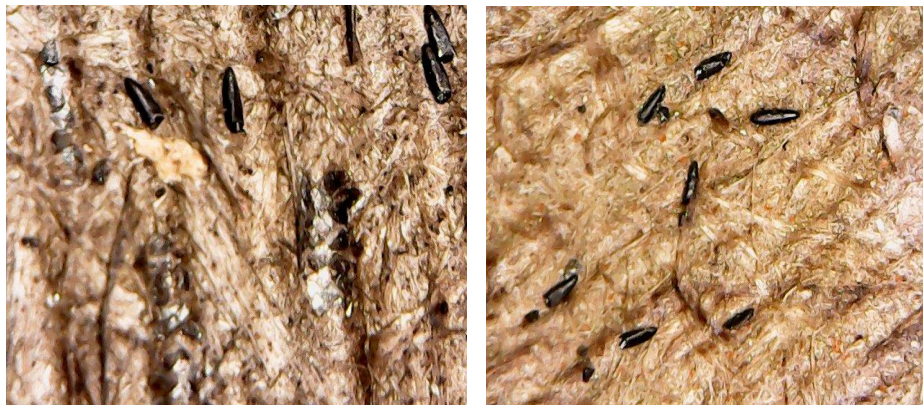


Figura 14 - Imagem com ampliação (esquerda) e imagem sem ampliação (direita) utilizadas como exemplo para o capítulo 3.7

### 3.2 Método 1

Para este método utilizou-se o primeiro conjunto de imagens descrito no capítulo 3.1. Como as imagens obtidas se encontravam no modelo RGB, e este modelo não contém informação relevante para a segmentação [3], a primeira tarefa realizada foi a conversão das imagens para outro modelo de cores. Após a realização de alguns testes comprovou-se que os componentes da tonalidade nos modelos de cores HSV e YIQ e o componente da luminosidade do modelo de cores  $L^*a^*b$  foram os que obtiveram os melhores resultados no processo de segmentação. Para este método o modelo escolhido foi o HSV, e como é possível perceber na Figura 15 (topo-esquerda), o componente da tonalidade (*Hue*) não retém muitas informações da *ovitrampa*.

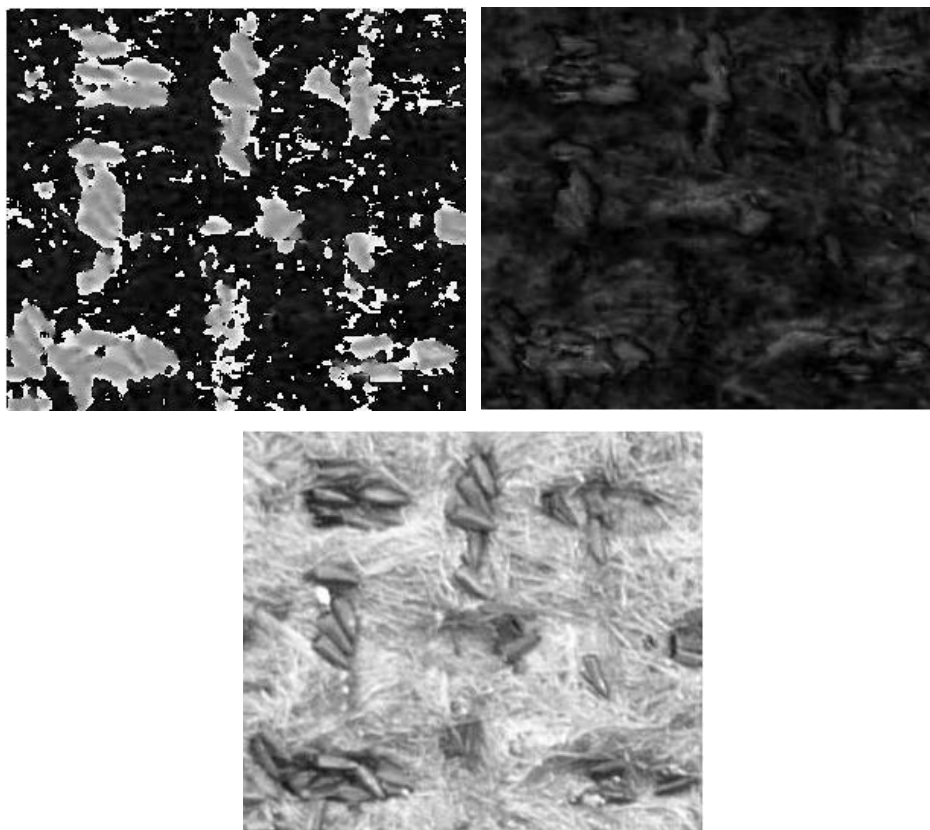


Figura 15 - Componentes do modelo HSV para a imagem da Figura 3: tonalidade (topo-esquerda), saturação (topo-direita) e valor (abaixo)

Um dos principais problemas para a criação de um método para a contagem automática de ovos em uma determinada imagem é torná-la um conjunto de objetos ou áreas que possam ser separadas por sua relevância, logo, operações de segmentação são extremamente importantes para o isolamento destes elementos [3]. Portanto, métodos de binarização foram utilizados para o processo de segmentação das imagens.

Neste caso, os métodos de binarização com abordagem global foram adotados e aplicados no componente da tonalidade (*Hue*) da imagem. Embora existam vários algoritmos com esta abordagem na literatura [7], estes possuem uma grande dificuldade quanto ao desempenho, dado que suas performances dependem do conjunto de imagens. Assim sendo, dois métodos conhecidos da literatura foram escolhidos: o método proposto por *Otsu* e o método proposto por *Kittler*.

No MATLAB® [4] a função *graythresh* utiliza (por padrão) o método de *Otsu*, enquanto que o método de *Kittler* não existe por padrão no software, porém é facilmente encontrado na internet, sendo necessário somente sua importação. O resultado da aplicação destes métodos na tonalidade da imagem pode ser visto na Figura 16.

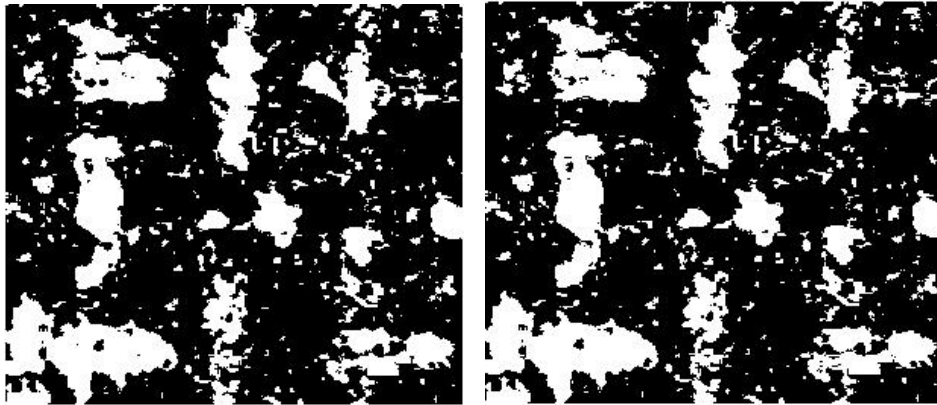


Figura 16 - Componente da tonalidade após a binarização com o método de *Otsu* (esquerda) e *Kittler* (direita)

Tendo em mãos as imagens geradas pelos métodos, a próxima tarefa foi aplicar um algoritmo de rotulação de componentes conexos [3]. Este algoritmo coloca um rótulo único para cada área da imagem (neste caso, as áreas em branco na Figura 16), fazendo com que cada uma destas áreas possa ser avaliada individualmente. No software MATLAB® [4] a função *bwlabel* fica responsável por realizar esta tarefa. A rotulação dos componentes simplificou a tarefa de eliminar pequenas áreas que não se caracterizavam como ovos do mosquito. Desta forma, áreas que não continham um valor superior a 170 pixels foram descartadas produzindo as imagens resultantes apresentadas na Figura 17.

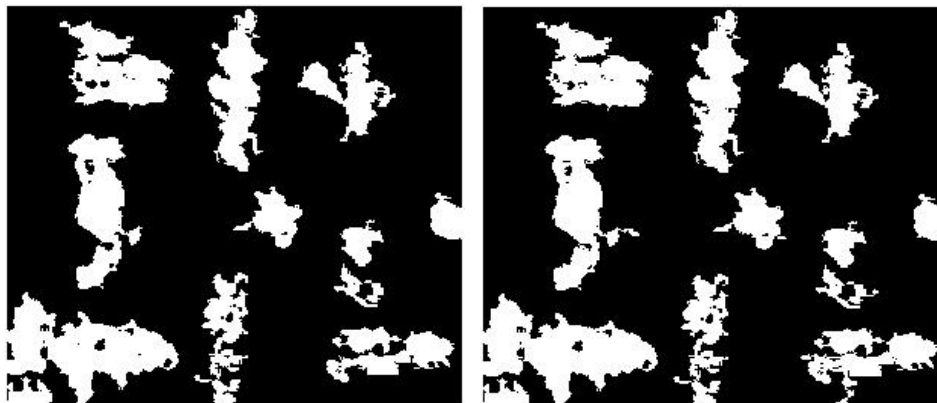


Figura 17 - Eliminação das pequenas áreas da imagem binarizada pelo método de *Otsu* (esquerda) e *Kittler* (direita)

Em seguida, as imagens passaram por um processo de filtragem para remover imperfeições e retornar informações significativas para a contagem. O filtro aplicado nesta etapa foi a operação morfológica de fechamento e sua utilização só é possível desde que se defina um elemento estruturante. Para a definição deste elemento dois parâmetros foram levados em consideração: o tamanho do ovo e a sua inclinação, visto que os ovos possuem



tamanhos e inclinações diferentes. Levando-se em consideração estes parâmetros, ficou decidido que o elemento estrutural teria uma pequena inclinação e seu tamanho seria de aproximadamente 270 pixels, valor esse que corresponde ao tamanho médio de um ovo. Para realizar a operação morfológica no MATLAB® [4] duas funções são necessárias: a *imerode* e a *imdilate*; A primeira é responsável pela operação de erosão e a segunda pela operação de dilatação.

A Figura 18 apresenta o elemento estruturante (que ficou com uma resolução 18 x 30 pixels) e a Figura 19 mostra o resultado da execução da operação morfológica onde as áreas que antes possuíam imperfeições agora se encontram mais delimitadas.

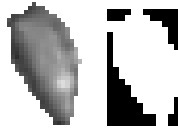


Figura 18 - Ovo médio (esquerda) utilizado para a definição do elemento estruturante (direita)



Figura 19 - Imagem resultante da operação morfológica para as imagens binarizadas pelo método de *Otsu* (esquerda) e *Kittler* (direita)

Além da remoção das imperfeições, pode-se notar na imagem acima que as áreas agora possuem um formato de ovo similar aos ovos encontrados na imagem original. Finalmente, a quantidade total de ovos foi obtida pela seguinte fórmula:

$$N^{\circ} \text{ total de ovos} = \frac{N^{\circ} \text{ total de pixels brancos}}{\text{Tamanho médio de um ovo}}$$

Portanto, o total de ovos da imagem foi calculado dividindo-se o número total de pixels brancos contidos na imagem pelo tamanho médio de um ovo de mosquito (270 pixels). O fluxo a seguir resume as etapas executadas neste método:



Figura 20 - Fluxo de execução do método 1

### 3.3 Método 2

Nesta abordagem utilizou-se, assim como no primeiro método, o primeiro conjunto de imagens do capítulo 3.1 e converteram-se as imagens do formato RGB para o modelo de cores YIQ. Também como no método anterior o componente da tonalidade (neste caso o componente *I*) foi escolhido para ser segmentado e o método de binarização adotado nesta etapa foi o *threshold* fixo. Nos testes realizados verificou-se que o ponto de corte com um valor 200 reproduziu o melhor resultado, como pode ser visto na Figura 21.

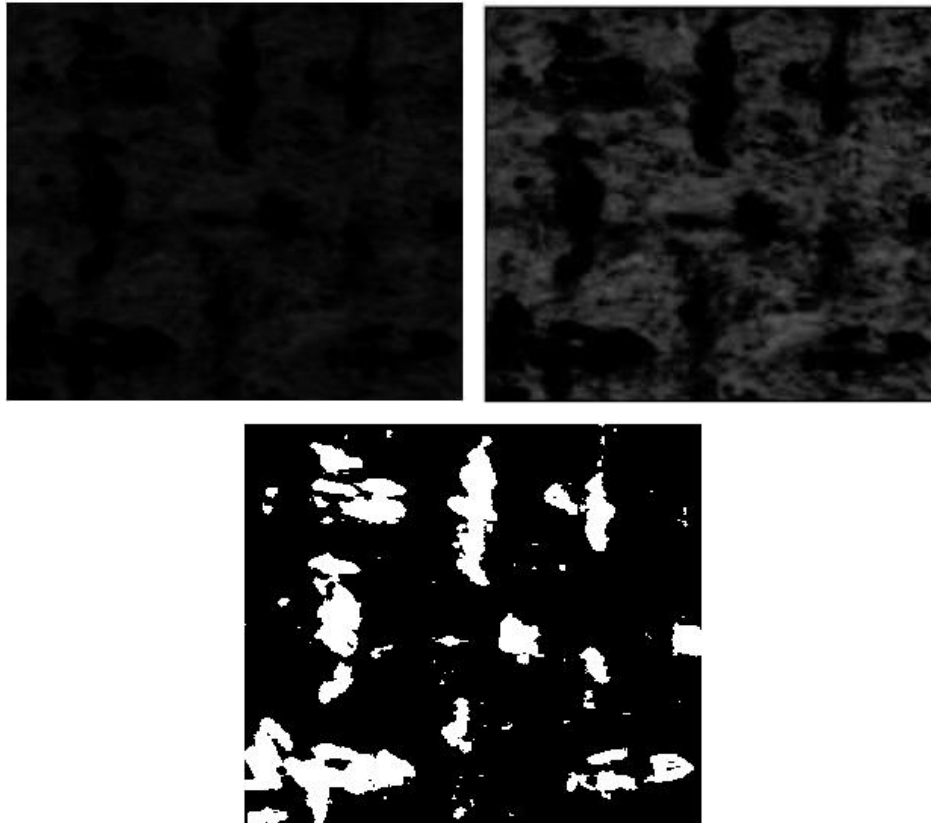


Figura 21 - Componente I do modelo YIQ (topo-esquerda), imagem alterada para fácil visualização das diferentes áreas (topo-direita) e a binarização com threshold fixo (abaixo-esquerda)

Como podemos ver na Figura 21 (topo-esquerda) o componente I do modelo retorna uma imagem muito escura, dificultando para o leitor a visualização das diferentes áreas. Com o intuito de melhorar a inspeção da imagem pelo leitor, a Figura 21 (topo-esquerda) foi alterada adicionando-a mais luminosidade como é apresentado na Figura 21 (topo-direita).

Os passos seguintes foram os mesmos aplicados ao método 1: executou-se o algoritmo de rotulação nas imagens, eliminaram-se as áreas que não continham tamanho superior a 170 pixels (Figura 22 - esquerda), aplicou-se a operação morfológica de fechamento com o elemento estruturante (Figura 22 - direita) e finalmente calculou-se a quantidade de ovos dividindo-se o valor total de pixels brancos pelo tamanho de um ovo médio. Um fluxo das etapas executadas pelo método 2 pode ser visto na Figura 23.

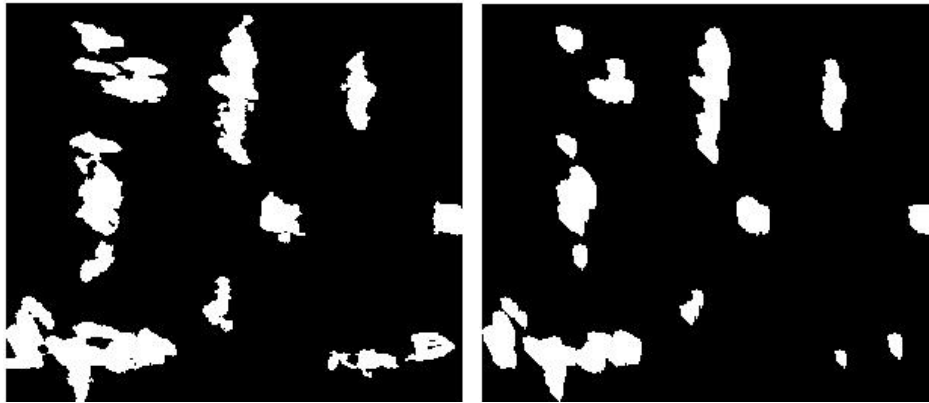


Figura 22 - Remoção das áreas inferiores a 170 *pixels* e aplicação da operação morfológica de fechamento



Figura 23 - Fluxo de execução do método 2

### 3.4 Método 3

Para este terceiro método, adotou-se o modelo de cores  $L^*a^*b$  e o componente da luminosidade ( $L$ ) foi escolhido como *input* para o processo de segmentação. Assim como no método 2, a binarização com *threshold* fixo foi aplicada (Figura 24 – abaixo) com o valor 0 para o ponto de corte. As etapas posteriores foram as mesmas executadas nos métodos 1 e 2: executou-se o algoritmo de rotulação nas imagens, eliminaram-se as áreas que não continham tamanho superior a 170 *pixels*, aplicou-se a operação morfológica de fechamento com o elemento estruturante e finalmente calculou-se a quantidade de ovos dividindo-se o valor total de *pixels* brancos pelo tamanho de um ovo médio.

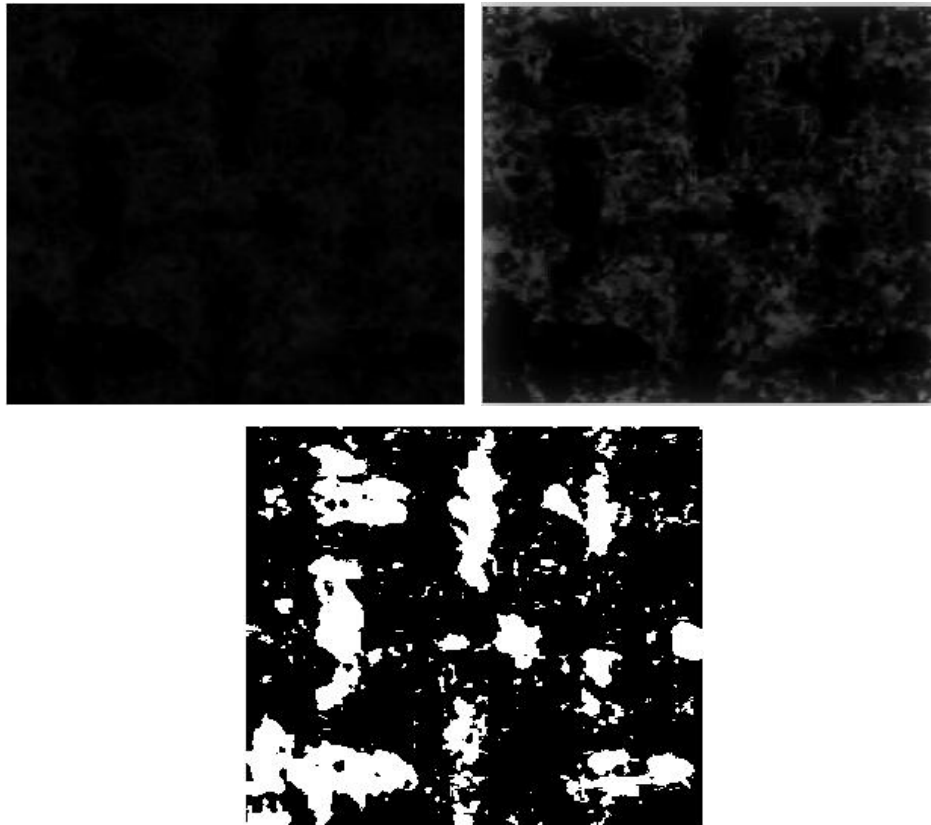


Figura 24 - Componente da luminosidade (L) do modelo L\*a\*b\* (topo-esquerda), imagem alterada para fácil visualização das diferentes áreas (topo-direita) e a binarização com threshold fixo (abaixo)

Como o componente da luminosidade retorna uma imagem muito escura, dificultando a visualização das diferentes áreas, ela foi alterada resultando na Figura 24 (topo-direita). A figura a seguir mostra o fluxo de execução para este método.

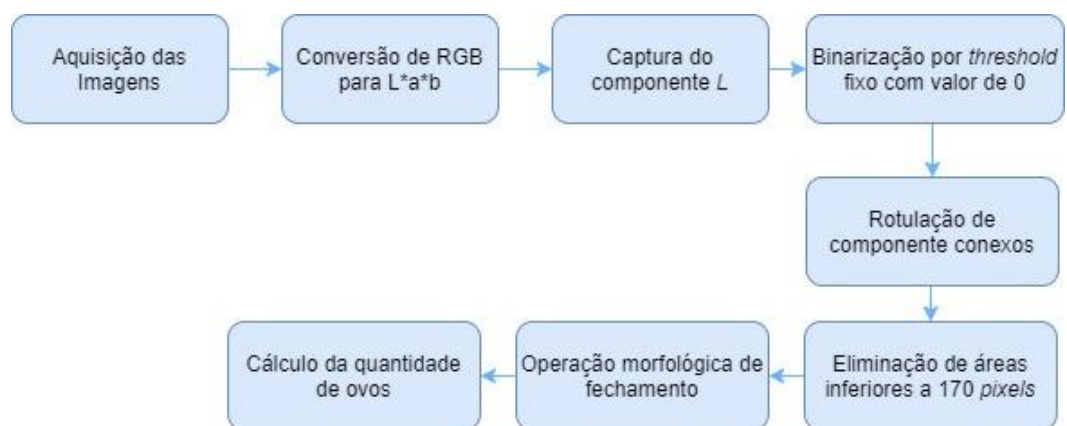


Figura 25 - Fluxo de execução para o método 3

### 3.5 Contagem de Ovos para Diferentes Inclinações do Elemento Estruturante

Este método foi desenvolvido para verificar a confiabilidade dos métodos anteriores e de etapas executadas neles. O objeto de investigação foi a inclinação adotada para o elemento estruturante e se esta inclinação poderia influenciar diretamente nos resultados gerados.

Inicialmente o método rotaciona o elemento estruturante para ângulos entre 0 e 360 graus, variando estes ângulos a cada 1 grau. As rotações foram feitas pela função *imrotate* do MATLAB® [4]. Para cada uma destas rotações aplicou-se os métodos de contagem (desenvolvidos nos capítulos 3.2, 3.3 e 3.4) e os valores gerados foram armazenados. Ao fim de todo este processo calculou-se a média e o desvio padrão dos dados produzidos por cada método de contagem. Com o objetivo de produzir uma maior quantidade de informações para posteriores análises, decidiu-se também variar os ângulos a cada: 2, 3, 4, 5, 6, 10 e 15 graus.

### 3.6 Contagem de Ovos em Divisões de Imagens de *Ovitrampas*

Este método teve como propósito avaliar o desempenho dos algoritmos de contagem para divisões das imagens do primeiro conjunto, visando a aplicação destes algoritmos em imagens completas da *ovitrampa*, após a captura com a câmera fotográfica. Um exemplo de imagem completa da pode ser visto na imagem abaixo:



Figura 26 - Exemplo de uma *ovitrampa* completa

Este método inicia-se com a captura do tamanho da imagem, ou seja, os valores que correspondem a sua largura e comprimento. Para o entendimento de como foi dividida a imagem, a partir dos seus valores de largura e comprimento, vamos utilizar uma matriz quadrada como analogia. Uma matriz quadrada é uma matriz que possui quantidade de linhas

igual a quantidade de colunas, portanto uma matriz quadrada de ordem 3 será tomada como exemplo.

Para representar a imagem como uma matriz 3 x 3 divide-se inicialmente a sua largura e comprimento pela ordem da matriz, para este caso a ordem é 3. O resultado desta divisão informa o valor necessário para retirar da largura e do comprimento afim de obter um elemento da matriz, ou seja, uma divisão da imagem. Um exemplo deste processo pode ser visto na Figura 27.

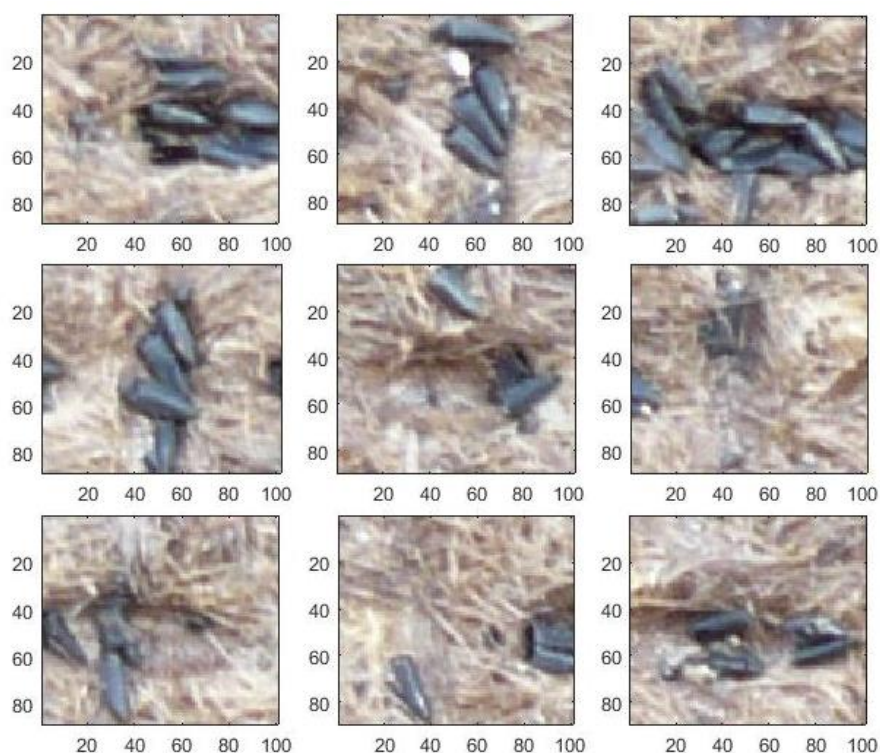


Figura 27 – Processo de divisão da Figura 3 como uma matriz quadrada de ordem 3

A imagem acima possui um tamanho de 302 x 264 *pixels*, após a divisão, cada parte da imagem ficou com um tamanho de aproximadamente 100 x 88 *pixels*. As imagens foram cortadas com a função *imcrop* do MATLAB® [4].

A próxima etapa foi aplicar os métodos de contagem para cada divisão e os valores de retorno foram somados, resultando no total de ovos da imagem. Com o objetivo de produzir uma maior quantidade de informações para posteriores análises, variou-se a ordem da matriz quadrada de 1 (imagem sem divisão) até 10.

### 3.7 Contagem de Ovos para um Novo Conjunto de Imagens

Este último método teve como objetivo utilizar os métodos de contagem desenvolvidos (dos capítulos 3.2 ao 3.4) para o segundo conjunto de imagens, descrito no

capítulo 3.1. Inspeccionando as imagens deste conjunto percebeu-se que algumas possuíam uma maior ampliação do que outras. Como os métodos utilizam o tamanho médio de um ovo na última etapa do processo de contagem, esta ampliação poderia influenciar na decisão deste tamanho, portanto, optou-se por dividir as imagens em dois grupos: imagens com ampliação (6 imagens) e imagens sem ampliação (13 imagens). Portanto, este capítulo será dividido em duas partes; a primeira parte irá tratar das imagens com ampliação e a Figura 14 (esquerda, página 22) será utilizada como exemplo para a fácil compreensão das etapas, enquanto que a segunda parte irá ocupar-se das imagens sem ampliação e utilizará a Figura 14 (direita, página 22) como exemplo.

### 3.7.1 Contagem para Imagens com Ampliação

Os métodos deste e do próximo capítulo tiveram que realizar ajustes iniciais nas imagens. Confrontando-se as imagens da Figura 14 e a imagem da Figura 13, é perceptível a diferença de luminosidade entre as imagens e esta luminosidade impede a eficácia dos métodos desenvolvidos. Deste modo, funções do MATLAB® [4] foram necessárias para a concretização desta tarefa. Para isto, duas funções foram utilizadas: a *imadjust* e a *stretchlim*. A primeira função retorna uma imagem após o ajuste nos valores de intensidade ou mapa de cores e seus parâmetros de entrada são, nesta ordem:

- A imagem;
- Um vetor, com valores entre 0 e 1, que representa os limites de contraste que serão aplicados na imagem de entrada;
- Um vetor, com valores entre 0 e 1, que representa os limites de contraste que serão aplicados na imagem de saída;
- Formato da curva que descreve a relação entre os valores de entrada e saída.

Se a imagem passada como parâmetro estiver no formato RGB, a função irá aplicar o mapeamento para os três componentes. Já a função *stretchlim* retorna os limites de contraste da imagem passada como parâmetro e seus parâmetros de entrada são: a imagem e o valor da fração da imagem para saturar os valores dos limites de contraste.

Após combinar estas funções e aplicá-la na imagem, os passos seguintes foram os mesmos listados nos capítulos anteriores, salvo algumas pequenas modificações; A primeira modificação foi no valor definido para a eliminação das pequenas áreas, que teve um aumento de 170 pixels para 430 pixels. A segunda foi no aumento do tamanho do ovo médio, visto que



os tamanhos dos ovos eram maiores do que no primeiro conjunto de imagens. Pelos testes realizados, um ovo médio neste conjunto de imagens tem aproximadamente 2200 pixels.

A terceira modificação foi no processo de filtragem, onde removeu-se a função *imerode* e utilizou-se unicamente a função *imdilate*. E finalmente, a última modificação realizada foi no cálculo da contagem de ovos, ao trocar-se o valor de 270 pixels para 2200 pixels. As imagens da Figura 28 mostram os efeitos causados pelas modificações realizadas tomando como exemplo o método de contagem do capítulo 3.3.

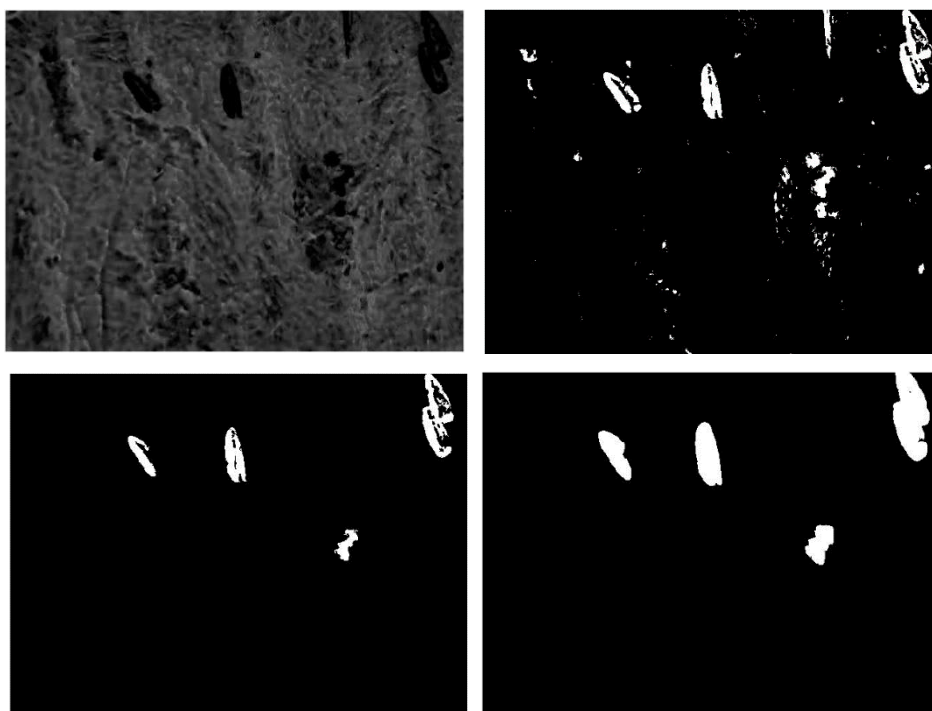


Figura 28 - Componente I do modelo YIQ (topo-esquerda) da imagem da Figura 4 (esquerda), binarização com threshold fixo (topo-direita), remoção de pequenas áreas (baixo-esquerda) e aplicação da função *imdilate* (baixo-direita)

Como pode ser visto nas imagens acima, o método proposto retorna os ovos presentes na *ovitrampa* e também consegue eliminar grande parte das áreas que não se caracterizam como ovos.

### 3.7.2 Contagem para Imagens sem Ampliação

Este método utilizou os mesmos passos do método anterior, salvo também pequenas alterações; O valor definido para a eliminação das pequenas áreas, que teve um aumento de 170 pixels para 320 pixels e o tamanho do ovo médio que, pelos testes realizados, tem aproximadamente 1475 pixels. As imagens da Figura 29 mostram os efeitos causados pelas modificações realizadas tomando como exemplo o método de contagem do capítulo 3.3.

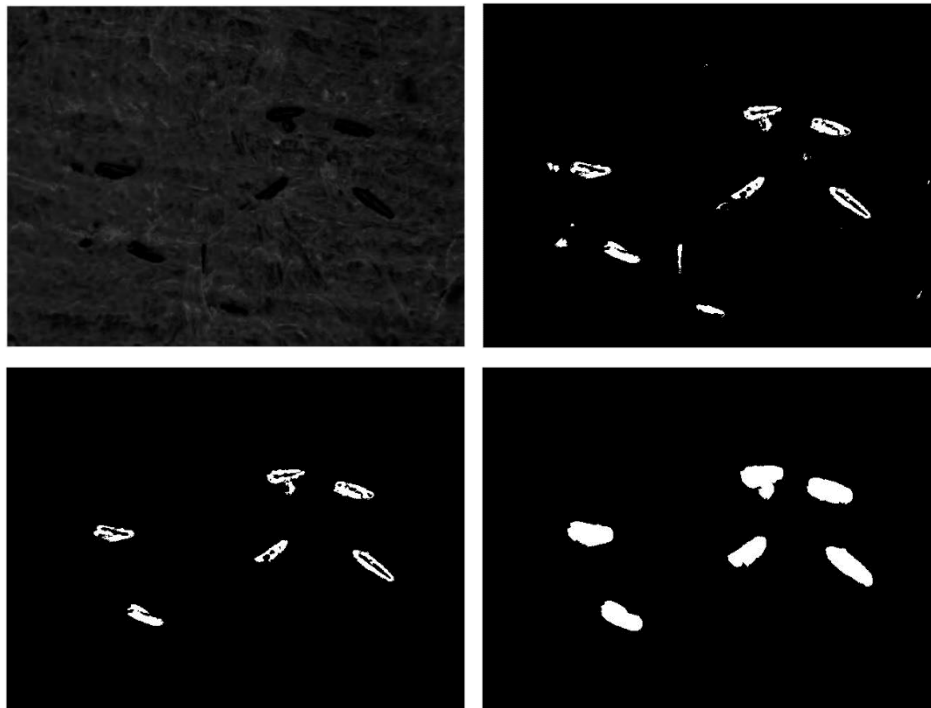


Figura 29 – Componente I do modelo YIQ (topo-esquerda) da imagem da Figura 4 (direita), binarização com threshold fixo (topo-direita), remoção de pequenas áreas (baixo-esquerda) e aplicação da função *imdilate* (baixo-direita)

Como pode ser visualizado nas imagens acima o método retornou todos os ovos presentes na imagem e conseguiu eliminar as áreas que não se caracterizava como ovos.

## 4 Resultados Experimentais

Este capítulo apresenta os experimentos e as avaliações executadas sobre os métodos desenvolvidos nos capítulos precedentes.

### 4.1 Experimento 1

O primeiro experimento foi realizado para avaliar os resultados obtidos pelos métodos desenvolvidos dos capítulos 3.2 ao 3.4. Para isto, inicialmente executou-se os métodos sobre o primeiro conjunto de imagens (descrito no capítulo 3.1) e os resultados gerados foram comparados com a quantidade de ovos que cada imagem contém.

Como o método do capítulo 3.2 possui dois algoritmos de binarização, decidiu-se separar os valores retornados em dois grupos: um grupo para o algoritmo de *Otsu* e outro para o algoritmo de *Kittler*. Afim de comparar os valores obtidos pelos métodos, o erro médio foi adotado como métrica. O cálculo do erro médio para um conjunto de medidas é dado pela seguinte fórmula:

$$\text{Erro médio} = \frac{(\sum \text{Erro Absoluto})}{n^{\circ} \text{ de medidas}}$$

$$\text{Erro absoluto} = \frac{|\text{Valor medido} - \text{Valor real}|}{\text{Valor real}}$$

Portanto, o erro médio é a soma do erro absoluto de cada medida do conjunto de dados, dividido pela quantidade total de elementos deste conjunto. A Tabela 1 mostra os resultados obtidos pelos métodos 1, 2 e 3 e seus respectivos erros médio.

Tabela 1 – Resultado da contagem para os métodos desenvolvidos dos capítulos 3.2 ao 3.4

Imagem	Quantidade de ovos na imagem	Quantidade estimada de ovos			
		Método 1		Método 2	Método 3
		<i>Otsu</i>	<i>Kittler</i>		
1	3	4	4	4	4
2	8	10	10	9	10
3	19	19	21	22	19
4	22	23	26	23	25
5	30	26	28	30	26
6	34	33	34	34	34

7	111	114	116	118	112
Erro Médio %		11,69	14	10	12,31

É possível verificar que os valores de erro médio são admissíveis em comparação com métodos não-automáticos (erro médio de até 20%), entretanto o método 2 obteve o melhor resultado para o experimento.

Após obter os valores de erro médio com os métodos propostos é importante compará-los com trabalhos existentes na literatura. Para esta tarefa foram utilizadas unicamente as imagens: 2, 3, 4, 5 e 7 da Tabela 1, pois os métodos da literatura só possuíam resultados para estas imagens em seus experimentos. Logo, os erros médio para os métodos do artigo de referência [3] foram:

- 10,39% para o método 1;
- 15,16% para o método 2;
- 12,03% para o método 3.

Enquanto que nos métodos desenvolvidos neste trabalho foram:

- 9,11% para o método 1 com *Otsu*;
- 12,97% para o método 1 com *Kittler*;
- 7,82% para o método 2;
- 10,57% para o método 3.

Pode-se verificar que tanto na primeira análise quanto nesta, o método 2 deste trabalho continua retornando o melhor resultado.

## 4.2 Experimento 2

Este experimento foi realizado para avaliar os resultados gerados pelo método desenvolvido no capítulo 3.5. Nesse capítulo foi mostrado que os métodos retornaram a quantidade de ovos para um determinada rotação do elemento estruturante e a partir destes valores calculou-se as médias e os desvio padrão para cada método de contagem. As Tabelas 2, 3, 4 e 5 mostram os estes resultados para imagens que continham 8, 19, 22, 34 e 111 ovos respectivamente.

Tabela 2 - Resultado da contagem rotacionando o elemento estruturante para uma imagem com 8 ovos.

Variação dos ângulos	Método 1		Método 2		Método 3			
	<i>Otsu</i>		<i>Kittler</i>		Média	Desvio padrão	Média	Desvio padrão
	Média	Desvio padrão	Média	Desvio padrão				
1 em 1 grau	9,111	0,707	10,100	0,589	8,222	0,736	9,119	0,712
2 em 2 graus	9,128	0,701	10,122	0,585	8,228	0,739	9,144	0,694
3 em 3 graus	9,150	0,706	10,117	0,568	8,225	0,727	9,083	0,740
4 em 4 graus	9,200	0,657	10,144	0,591	8,233	0,735	9,122	0,684
5 em 5 graus	9,069	0,657	10,083	0,524	8,250	0,746	9,097	0,715
10 em 10 graus	9,139	0,639	10,139	0,543	8,278	0,741	9,083	0,770
15 em 15 graus	9,042	0,690	10,083	0,504	8,333	0,702	8,958	0,690

Tabela 3 - Resultado da contagem rotacionando o elemento estruturante para uma imagem com 19 ovos

Variação dos ângulos	Método 1		Método 2		Método 3			
	<i>Otsu</i>		<i>Kittler</i>		Média	Desvio padrão	Média	Desvio padrão
	Média	Desvio padrão	Média	Desvio padrão				
1 em 1 grau	19,814	0,915	20,778	1,032	22,711	0,659	19,192	1,009
2 em 2 graus	19,833	0,912	20,806	1,042	22,717	0,679	19,244	1,050
3 em 3 graus	19,775	0,921	20,775	1,025	22,692	0,658	19,175	0,976
4 em 4 graus	19,811	0,860	20,767	1,006	22,711	0,691	19,256	1,045
5 em 5 graus	19,722	0,843	20,694	0,988	22,653	0,653	19,167	1,021
10 em 10 graus	19,750	0,874	20,778	1,072	22,639	0,639	19,167	1,056
15 em 15 graus	19,542	0,833	20,667	1,007	22,708	0,751	19,083	1,018

Tabela 4 - Resultado da contagem rotacionando o elemento estruturante para uma imagem com 22 ovos

Variação dos ângulos	Método 1		Método 2		Método 3			
	<i>Otsu</i>		<i>Kittler</i>		Média	Desvio padrão	Média	Desvio padrão
	Média	Desvio padrão	Média	Desvio padrão				
1 em 1 grau	23,019	1,881	26,533	2,175	23,706	1,480	23,511	2,204
2 em 2 graus	23,056	1,881	26,572	2,197	23,706	1,494	23,544	2,228
3 em 3 graus	23,017	1,842	26,508	2,169	23,725	1,443	23,483	2,223
4 em 4 graus	23,089	1,912	26,611	2,171	23,678	1,490	23,578	2,172
5 em 5 graus	22,972	1,861	26,403	2,134	23,667	1,454	23,361	2,112
10 em 10 graus	22,944	1,851	26,389	2,060	23,639	1,437	23,389	2,128
15 em 15 graus	22,917	1,816	26,208	2,105	23,958	1,334	23,125	2,153

Tabela 5 - Resultado da contagem rotacionando o elemento estruturante para uma imagem com 34 ovos

Variação dos ângulos	Método 1		Método 2		Método 3			
	<i>Otsu</i>		<i>Kittler</i>		Média	Desvio padrão	Média	Desvio padrão
	Média	Desvio padrão	Média	Desvio padrão				
1 em 1 grau	32,342	1,783	34,147	2,125	34,025	1,460	32,047	1,847
2 em 2 graus	32,367	1,806	34,167	2,118	34,039	1,489	32,128	1,837
3 em 3 graus	32,292	1,812	34,092	2,138	34,017	1,478	32,075	1,861
4 em 4 graus	32,344	1,837	34,189	2,130	34,022	1,476	32,189	1,866
5 em 5 graus	32,153	1,750	34,042	2,106	34,042	1,467	31,958	1,842
10 em 10 graus	32,222	1,807	34,056	2,177	34,056	1,530	32,139	1,759
15 em 15 graus	32,042	1,732	33,958	2,116	34,208	1,560	31,917	1,840

Tabela 6 - Resultado da contagem rotacionando o elemento estruturante para uma imagem com 111 ovos

Variação dos ângulos	Método 1		Método 2		Método 3			
	<i>Otsu</i>		<i>Kittler</i>		Média	Desvio padrão	Média	Desvio padrão
	Média	Desvio padrão	Média	Desvio padrão				
1 em 1 grau	112,12	2,85	115,00	2,87	120,98	3,39	111,69	2,60
2 em 2 graus	112,22	2,81	115,11	2,84	121,01	3,42	111,80	2,55
3 em 3 graus	112,11	3,01	114,97	3,03	120,95	3,31	111,62	2,69
4 em 4 graus	112,28	2,83	115,21	2,88	120,92	3,53	111,89	2,62
5 em 5 graus	111,86	2,98	114,74	2,96	120,89	3,30	111,43	2,68
10 em 10 graus	111,67	3,01	114,58	2,93	120,78	3,23	111,42	2,69
15 em 15 graus	111,79	3,45	114,75	3,37	121,13	3,18	111,21	2,87

A partir dos resultados obtidos é possível verificar que em todas as tabelas os valores sempre giram em torno da quantidade de ovos que as imagens contêm. A única discrepância encontrada foi na Tabela 6, onde o método 2 retornou (em média) valores de aproximadamente 120 ovos para uma imagem que continha um total de 111. Esta discrepância pode estar associada ao valor que o método retorna para esta imagem, visto que este método retornou o maior valor dentre os outros métodos (vide Tabela 1). Portanto, pode-se inferir que a inclinação do elemento estruturante não tem influência direta nos resultados gerados pelos métodos.

### 4.3 Experimento 3

Este experimento foi realizado para avaliar os resultados gerados pelo método desenvolvido no capítulo 3.6. Para a realização deste experimento duas imagens do primeiro conjunto de imagens foram selecionadas, pois possuíam os maiores tamanho e quantidade de ovos dentre outras. As imagens selecionadas foram: a imagem da Figura 13 (página 22) e a imagem da Figura 30, que possuem uma quantidade total de 34 e de 111 ovos, respectivamente.



Figura 30 - Imagem da *ovitrampa* contendo 111 ovos

Os resultados do experimento para a Figura 13 são apresentados na Tabela 7, enquanto que os resultados para a Figura 30 se encontram na Tabela 8.

Tabela 7 - Resultado das divisões da imagem da Figura 3

Ordem da matriz	Nº de divisões da imagem	Quantidade ovos na imagem	Quantidade estimada de ovos			
			Método 1		Método 2	Método 3
			<i>Otsu</i>	<i>Kittler</i>		
1	1	34	33	34	34	34
2	4	34	32	33	34	34
3	9	34	33	34	34	33
4	16	34	32	33	33	33
5	25	34	34	35	33	36
6	36	34	33	34	32	33
7	49	34	36	39	33	31
8	64	34	35	35	28	36
9	81	34	35	38	26	35
10	100	34	30	31	24	30

Tabela 8 - Resultado das divisões da imagem da Figura 16.

Ordem da matriz	Nº de divisões da imagem	Quantidade de ovos na imagem	Quantidade estimada de ovos			
			Método 1	Método 2	Método 3	
			<i>Otsu</i>	<i>Kittler</i>		
1	1	111	114	116	118	112
2	4	111	113	115	116	111
3	9	111	116	117	114	113
4	16	111	112	115	114	110
5	25	111	115	119	112	112
6	36	111	113	114	110	110
7	49	111	111	114	110	111
8	64	111	109	115	106	106
9	81	111	115	117	108	109
10	100	111	109	114	102	108

Analisando a Tabela 7 é possível verificar que se dividirmos a imagem em até 36 partes, todos os métodos retornam valores iguais ou próximos do valor esperado, já se for extrapolado este valor, alguns métodos chegam a retornar valores com diferenças de até 10 ovos (método 2 com ordem da matriz 10). Já para a Tabela 8 pode-se verificar que com a matriz de ordem 7 os métodos retornaram os melhores resultados, porém se passarmos deste valor discrepâncias também são encontradas. Então, pode-se concluir que os métodos se comportam bem com até 36 divisões, porém se passarmos destes valores discrepâncias podem ser encontradas.

#### 4.4 Experimento 4

Este último experimento foi realizado para avaliar os resultados gerados pelos métodos desenvolvidos no capítulo 3.7. No processo de desenvolvimento destes métodos, foi visto que as imagens tiveram que passar por ajustes e para isto foram utilizadas as seguintes funções do MATLAB® [4]: *imadjust* e a *stretchlim*. Nos experimentos realizados cada método recebeu uma configuração diferente para estas funções, como mostrado a seguir:

- Método 1 com *Otsu*:
  - *stretchlim*:
    - Valor da fração de saturação: 0.5;
  - *imadjust*:
    - Limites de contraste para imagem de entrada: [0.135 0 0; 1 1 1];



- Limites de contraste para imagem de saída: retorno da *stretchlim*;
  - Formato da curva: 0.1;
- Método 1 com *Kittler*:
  - *stretchlim*:
    - Valor da fração de saturação: 0.5;
  - *imadjust*:
    - Limites de contraste para imagem de entrada: [0.145 0 0; 1 1 1];
    - Limites de contraste para imagem de saída: retorno da *stretchlim*;
    - Formato da curva: 0.05.
- Método 2:
  - *stretchlim*:
    - Valor da fração de saturação: 0.5;
  - *imadjust*:
    - Limites de contraste para imagem de entrada: [0.11 0 0; 0.9 0.95 1];
    - Limites de contraste para imagem de saída: retorno da *stretchlim*;
    - Formato da curva: 0.5.
- Método 3:
  - *stretchlim*:
    - Valor da fração de saturação: 0.5;
  - *imadjust*:
    - Limites de contraste para imagem de entrada: [0.135 0.135 0; 1 0.6 1];
    - Limites de contraste para imagem de saída: retorno da *stretchlim*;
    - Formato da curva: 1.

Com os métodos devidamente configurados, a próxima etapa foi executá-los sobre o segundo conjunto de imagens (descrito no capítulo 3.1) e os resultados obtidos foram comparados com a quantidade de ovos que cada imagem tinha. Como o capítulo 3.7 foi dividido para os conjuntos de imagens com ampliação e sem ampliação, optou-se por separar os resultados obtidos em duas tabelas distintas. Logo, a Tabela 9 mostra os resultados obtidos sobre as imagens com ampliação, enquanto que a Tabela 10 mostra os resultados para as imagens sem resolução.

Tabela 9 - Resultado da contagem para os métodos desenvolvidos dos capítulos 3.7 para imagens com ampliação

Imagem	Quantidade de ovos na imagem	Quantidade estimada de ovos			
		Método 1		Método 2	Método 3
		<i>Otsu</i>	<i>Kittler</i>		
1	2	4	6	2	4
2	4	5	6	4	7
4	5	7	9	6	7
5	5	8	9	6	7
6	6	6	9	5	7
Erro médio %		45	92	11,33	54

Tabela 10 - Resultado da contagem para os métodos desenvolvidos dos capítulos 3.7 para imagens sem ampliação

Imagem	Quantidade de ovos na imagem	Quantidade estimada de ovos			
		Método 1		Método 2	Método 3
		<i>Otsu</i>	<i>Kittler</i>		
1	1	0	2	2	2
2	2	8	207	2	3
3	3	3	3	3	3
4	4	3	4	3	4
5	5	5	5	5	5
6	5	5	8	4	5
7	6	6	8	6	7
8	7	6	9	6	6
9	7	7	8	6	7
10	7	6	7	6	6
11	8	11	14	8	8
12	12	11	12	10	9
13	13	9	12	9	10
Erro médio %		38,41	840	15,73	31,04

Na Tabela 7 observa-se que os métodos 1 e 3 retornam valores muito próximos ao esperado, porém seus erros médios não são admissíveis em comparação com métodos de contagem não automáticos. Somente o método 2 retornou valores dentro do limite tolerável (até 20%), sendo ele o melhor método para este conjunto de imagens.

Na Tabela 8, apesar de retornarem valores muito próximos da quantidade de ovos da imagem (salvo a método 1 com *Kittler* para imagem 2 que será comentado logo após), os

métodos 1 e 3 não conseguiram atingir um erro médio admissível em comparação com métodos de contagem não automáticos. Entretanto, o método 2 apresentou o melhor resultado para o conjunto de imagens sem ampliação com seu erro médio dentro do limite admissíveis.

É possível perceber na Tabela 10, para a imagem 2, que o método 1 com a binarização de *Kittler* retornou um valor extremamente discrepante. A imagem utilizada que retornou este valor é apresentada a seguir:



Figura 31 - Imagem que gerou discrepância no resultado do método 1 com *Kittler*

Pode-se notar que esta imagem possui uma luminosidade bastante acentuada e mesmo com as configurações adotadas para o método, ela não foi totalmente eliminada. A imagem a seguir mostra o resultado da aplicação da binarização de *Kittler* na imagem após a aplicação dos ajustes de luminosidade.

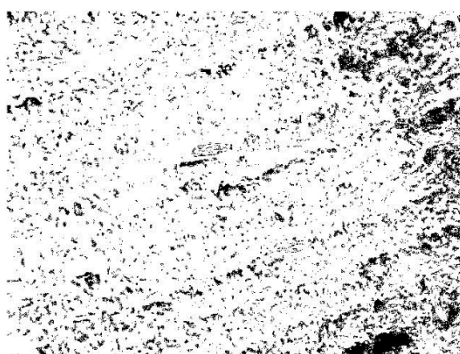


Figura 32 - Imagem após ajustes de luminosidade e aplicação da binarização de *Kittler*

Portanto, como o cálculo dos ovos depende da quantidade de áreas brancas, e a imagem acima está repleta delas, o método retornou um valor extremamente alto. Se esta imagem for retirada da análise e o erro médio para este método for recalculado, o novo valor será de 56,48%. Este valor ainda não é admissível em relação aos métodos de contagem não automático, porém já pode ser visto uma melhora significativa em relação ao valor calculado anteriormente.

## 5 Conclusões e Trabalhos Futuros

Este capítulo apresenta as considerações finais, as limitações que foram encontradas e trabalhos futuros.

### 5.1 Contribuições

Este trabalho propôs fazer:

- Fazer um estudo comparativo entre os métodos de contagem propostos na literatura e os métodos de contagem que foram desenvolvidos neste trabalho para um conjunto de imagens de *ovitrampas* capturadas através de uma câmera digital;
- Verificar a influência da inclinação do elemento estruturante utilizado nas operações morfológicas, etapa importante no processo de contagem automática;
- Avaliar o desempenho na contagem de ovos em divisões das imagens de *ovitrampas*;
- Aplicar os algoritmos de contagem deste trabalho, desenvolvidos para as imagens da câmera digital, em um novo conjunto de imagens obtidos através de um microscópio digital.

Para o primeiro objetivo constatou-se que os métodos obtiveram erros médios admissíveis em comparação com os métodos de contagem não automático, e também foi constatado que o método desenvolvido no capítulo 3.3 alcançou os melhores resultados em todas as avaliações. Para o segundo objetivo estipulado observou-se que os valores da média e desvio padrão giraram em torno da quantidade de ovos presentes nas imagens, chegando à conclusão de que a inclinação do elemento estruturante não influencia diretamente nos valores calculados pelos métodos.

Na avaliação do desempenho na contagem de ovos em divisões das imagens, comprovou-se que os métodos retornam valores admissíveis até um certo número de divisões e se este valor for extrapolado, os métodos poderão retornar valores discrepantes.

E finalmente para o cálculo de ovos para um novo conjunto de imagens, foi observado que ajustes foram necessários para a utilização dos métodos de contagens desenvolvidos e que somente o método 2 obteve um erro médio dentro dos limites admissíveis. Já os outros métodos obtiveram erros médios fora dos limites toleráveis, e isto está diretamente relacionado com a luminosidade presente nas imagens. Portanto, é necessário um maior refinamento nas configurações adotadas para cada método.

## 5.2 Trabalhos Futuros

Os seguintes trabalhos futuros foram identificados como sugestão para a continuação e melhoria deste trabalho:

- Utilização de outros métodos de binarização tanto locais como globais para um maior refinamento sobre quais técnicas teriam melhor desempenho
- Fazer uma avaliação mais apurada do método de divisão de imagens utilizando imagens completas de *ovitrampas*
- Buscar outras técnicas de ajuste para o novo conjunto de imagens, ou refinar os valores de configuração adotados.
- Buscar um conjunto maior de imagens para realizar mais experimentos
- Buscar um conjunto de imagens a respeito de uma palheta completa, pois foram utilizadas apenas imagens pontuais.

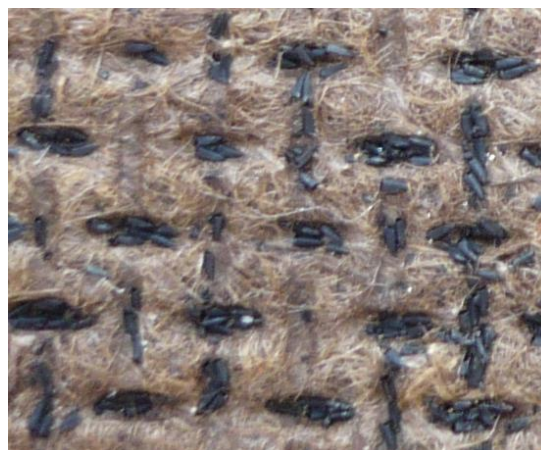
## Bibliografia

- [1] “Casos da dengue. Brasil, Grandes regiões e Unidades Federadas. 1990 a 2016”. [Online]. Disponível em: <http://portalsaude.saude.gov.br/index.php/situacao-epidemiologica-dados-dengue>. (Acesso em 8 de abril de 2017).
- [2] “Avaliação de armadilhas para a vigilância entomológica de *Aedes aegypti* com vistas à elaboração de novos índices de infestação.” [Online]. Disponível em: [http://www.fiocruz.br/ioc/media/nota\\_tecnica\\_ioc\\_3.pdf](http://www.fiocruz.br/ioc/media/nota_tecnica_ioc_3.pdf)
- [3] C.A.B. Mello, W.P. Santos, M.A.B. Rodrigues, A.L.B. Candeias, C.M.G. Gusmão. “Image Segmentation of Ovitrams for Automatic Counting of *Aedes Aegypti* Eggs.” in *30<sup>th</sup> Annu. International Conf. of the IEEE Engineering in Medicine and Biology Society*, Vancouver, p. 3103-3106, 2008.
- [4] Mathworks Inc. “*Student Edition of MATLAB Version 5 for Windows*”. Prentice Hall, Upper Saddle River, New Jersey, 1997.
- [5] O.M. Filho, H.V. Neto. “*Processamento digital de imagens*”. Rio de Janeiro: Brasport Livros e Multimídia, 1999. ISBN 8574520098
- [6] A.C. Leão “*Gerenciamento de cores para imagens digitais*”. Dissertação (Mestrado em Artes Visuais) - Curso de Mestrado em Artes Visuais, Escola de Belas Artes, Belo Horizonte. 135 f, 2005.
- [7] Puneet, Naresh Kumar Garg, “*Binarization Techniques used for Grey Scale Images*,” *International Journal of Computer Applications*, Vol. 71, No. 1, pp. 8-11, June 2013.
- [8] G. Gusmao, S. C. S. Machado, and M. A. B. Rodrigues, “*A new algorithm for segmenting and counting Aedes aegypti eggs in ovitraps*” in *Proc. Annual Int. Conf. IEEE Engineering in Medicine and Biology Society*, p. 6714–6717, 2009.
- [9] H. Pedrini, W.R. Schwartz, “*Análise de Imagens Digitais: Princípios e Aplicações*”, Thomson, 2008.

- [10] J. Schanda, “*Colorimetry: understand the CIE system*” Hoboken: Wiley, 2007.
- [11] D. Pascale “*A Review of RGB Color Spaces ...from xyY to R'G'B'*”. Canada: The BabelColor Company, p. 35, 2003.
- [12] N. Otsu, “A thresholding selection method from gray-scale histogram”, IEEE Transactions on System, Man, and Cybernetics 9, p. 62–66, 1979
- [13] J. Kittler, J. Illingworth, “Minimum error thresholding”, Pattern Recognition 19 (1) p. 41–47, 1986.
- [14] R.C. Gonzalez, R.E. Woods, “Processamento digital de imagens”, 3. ed. São Paulo: Pearson, p. 644, 2010
- [15] J.E.R. Queiroz, H.M. Gomes “Introdução ao Processamento Digital de Imagens” RITA 13(2): p. 11-42, 2006
- [16] M. Athimethphat, “*A Review on Global Binarization Algorithms for Degraded Document Images,*” AU Journal of Technology, vol. 14, p. 188-195, January 2011
- [17] J.R. Parker. “*Algorithms for Image Processing and Computer Vision*”, John Wiley and Sons, 1997.
- [18] “*Mosquito – Aedes aegypti*”. [Online]. Disponível em: <http://mosquito.saude.es.gov.br/aedes-aedypti>. (Acesso em 8 de abril de 2017).

# Apêndice A

## Anexo 1

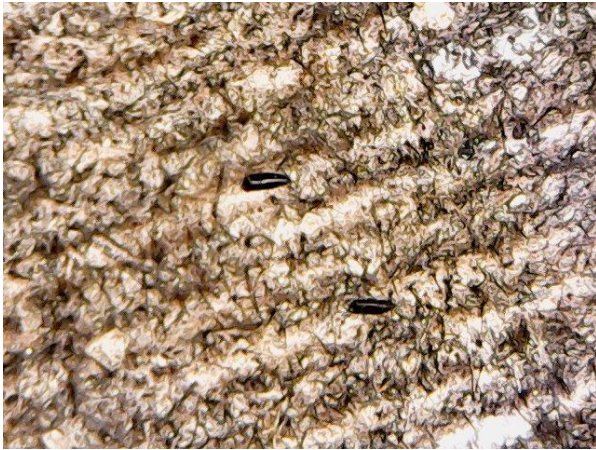
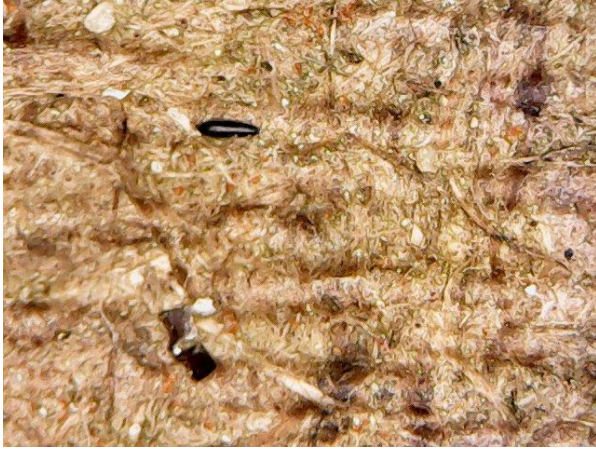




## Anexo 2







# Apêndice B

## Anexo 1

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% UNIVERSIDADE FEDERAL DE PERNAMBUCO %%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% CENTRO DE INFORMÁTICA %%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% DIOGENES DOS SANTOS SILVA %%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% 14/07/2017 %%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% MAIN %%%%%%%%%%%%%%
```

### % LEITURA DAS IMAGENS

```
ovos3 = 'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\3ovos.bmp';
ovos8 = 'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\8ovos.bmp';
ovos19 = 'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\19ovos.bmp';
ovos22 = 'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\22ovos.bmp';
ovos30 = 'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\30ovos.bmp';
ovos34 = 'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\34ovos.bmp';
ovos111 = 'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\111ovos.bmp';
```

### % LEITURA DO ELEMENTO ESTRUTURANTE

```
ovosel =
imread('C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\ovo_sel_8x13.bmp'
);
ovosell =
imread('C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\ovo_sel_18x30.bmp'
');
```

### % METODO PARA CALCULO DE OVOS COM A BINARIZACAO DE OTSU

```
display('##### Metodo 1 com Otsu threshold #####');
display(['Quantidade de ovos em (3 ovos) = ' int2str(HSV_Otsu(ovos3,
ovosell))] );
display(['Quantidade de ovos em (8 ovos) = ' int2str(HSV_Otsu(ovos8,
ovosell))] );
display(['Quantidade de ovos em (19 ovos) = ' int2str(HSV_Otsu(ovos19,
ovosell))] );
display(['Quantidade de ovos em (22 ovos) = ' int2str(HSV_Otsu(ovos22,
ovosell))] );
display(['Quantidade de ovos em (30 ovos) = ' int2str(HSV_Otsu(ovos30,
ovosell))] );
display(['Quantidade de ovos em (34 ovos) = ' int2str(HSV_Otsu(ovos34,
ovosell))] );
display(['Quantidade de ovos em (111 ovos) = ' int2str(HSV_Otsu(ovos111,
ovosell))] );
display('#####');
```

### % METODO PARA CALCULO DE OVOS COM A BINARIZACAO DE KITTLER

```
display('##### Metodo 1 com Kittler threshold #####');
display(['Quantidade de ovos em (3 ovos) = ' int2str(HSV_Kittler(ovos3,
ovosell))] );
display(['Quantidade de ovos em (8 ovos) = ' int2str(HSV_Kittler(ovos8,
ovosell))] );
display(['Quantidade de ovos em (19 ovos) = ' int2str(HSV_Kittler(ovos19,
ovosell))] );
display(['Quantidade de ovos em (22 ovos) = ' int2str(HSV_Kittler(ovos22,
ovosell))] );
display(['Quantidade de ovos em (30 ovos) = ' int2str(HSV_Kittler(ovos30,
ovosell))] );
display(['Quantidade de ovos em (34 ovos) = ' int2str(HSV_Kittler(ovos34,
ovosell))] );
```

```

display(['Quantidade de ovos em (111 ovos) = ' int2str(HSV_Kittler(ovos111,
ovosel1))]);
display('#####');

% METODO PARA CALCULO DE OVOS COM MODELO DE CORES YIQ E THRESHOLD FIXO
display('##### Metodo 2 com YIQ e threshold fixo #####');
display(['Quantidade de ovos em (3 ovos) = ' int2str(YIQ_Threshold(ovos3,
ovosel))]);
display(['Quantidade de ovos em (8 ovos) = ' int2str(YIQ_Threshold(ovos8,
ovosel))]);
display(['Quantidade de ovos em (19 ovos) = ' int2str(YIQ_Threshold(ovos19,
ovosel))]);
display(['Quantidade de ovos em (22 ovos) = ' int2str(YIQ_Threshold(ovos22,
ovosel))]);
display(['Quantidade de ovos em (30 ovos) = ' int2str(YIQ_Threshold(ovos30,
ovosel))]);
display(['Quantidade de ovos em (34 ovos) = ' int2str(YIQ_Threshold(ovos34,
ovosel))]);
display(['Quantidade de ovos em (111 ovos) = '
int2str(YIQ_Threshold(ovos111, ovosel))]);
display('#####');

% METODO PARA CALCULO DE OVOS COM MODELO DE CORES LAB E THRESHOLD FIXO
display('##### Metodo 2 com YIQ e threshold fixo #####');
display(['Quantidade de ovos em (3 ovos) = ' int2str(LAB(ovos3,
ovosel1))]);
display(['Quantidade de ovos em (8 ovos) = ' int2str(LAB(ovos8,
ovosel1))]);
display(['Quantidade de ovos em (19 ovos) = ' int2str(LAB(ovos19,
ovosel1))]);
display(['Quantidade de ovos em (22 ovos) = ' int2str(LAB(ovos22,
ovosel1))]);
display(['Quantidade de ovos em (30 ovos) = ' int2str(LAB(ovos30,
ovosel1))]);
display(['Quantidade de ovos em (34 ovos) = ' int2str(LAB(ovos34,
ovosel1))]);
display(['Quantidade de ovos em (111 ovos) = ' int2str(LAB(ovos111,
ovosel1))]);
display('#####');

```

## Anexo 2

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% UNIVERSIDADE FEDERAL DE PERNAMBUCO %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% CENTRO DE INFORMATICA %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% DIOGENES DOS SANTOS SILVA %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% 14/07/2017 %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% METODO 1 COM A BINARIZACAO DE OTSU %%%%%%%%%

```

```

function [total] = HSV_Otsu(imagem, ovo)

% LEITURA DA IMAGEM
rgb = imread(imagem);
[lin, col, plan] = size(rgb);

% COLOCANDO A IMAGEM EM FORMATO RGB, CASO NAO ESTEJA
if(size(rgb,3) == 1)
    rgb = cat(3, rgb, rgb, rgb);
end

```

```

% CONVERTENDO A IMAGEM PARA O MODELO DE CORES HSV
rgb = rgb2hsv(rgb);

% APLICANDO A BINARIZACAO DE OTSU
level = graythresh(rgb(:,:,1));
rgb = rgb(:,:,1) > level;

% APLICANDO O ALGORITMO DE COMPONENTES CONEXOS
[L, num] = bwlabel (rgb(:,:,1));
im_out = rgb(:,:,1);

% REMOVENDO AREAS INFERIORES A 270 PIXELS
for i=1:num
    elem = find(L == i);
    quant = size(elem);
    quant = quant(1);

    if (quant <= 270)
        im_out(elem) = 0;
    end
end

% APLICANDO A OPERACAO MORFOLOGICA DE FECHAMENTO
bw = imerode(im_out, ovo);
bw2 = imdilate(bw, ovo);
im_out = bw2;

% CONTANDO AS AREAS EM BRANCO E DIVIDINDO PELO TAMANHO DO OVO MEDIO
quant2 = find (bw2 == 1);
quant2 = size (quant2);
total = round(quant2(1)/270);
end

```

### Anexo 3

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% UNIVERSIDADE FEDERAL DE PERNAMBUCO %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% CENTRO DE INFORMATICA %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% DIOGENES DOS SANTOS SILVA %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% 14/07/2017 %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% METODO 1 COM A BINARIZACAO DE KITTLER %%%%%%%%%

function [total] = HSV_Kittler(imagem, ovo)

% LEITURA DA IMAGEM
rgb = imread(imagem);
[lin, col, plan] = size(rgb);

% COLOCANDO A IMAGEM EM FORMATO RGB, CASO NAO ESTEJA
if(size(rgb,3) == 1)
    rgb = cat(3, rgb, rgb, rgb);
end

% CONVERTENDO A IMAGEM PARA O MODELO DE CORES HSV
rgb = rgb2hsv(rgb);

% APLICANDO A BINARIZACAO DE KITTLER
% FUNCAO Kittler CRIADA POR BOB
% DISPONIVEL EM:

```

```

% https://www.mathworks.com/matlabcentral/fileexchange/45685-kittler-illingworth-thresholding
level = Kittler(rgb(:,:,1));
rgb = rgb(:,:,1) > level;

% APLICANDO O ALGORITMO DE COMPONENTES CONEXOS
[L, num] = bwlabel (rgb(:,:,1));
im_out = rgb(:,:,1);

% REMOVENDO AREAS INFERIORES A 270 PIXELS
for i=1:num
    elem = find(L == i);
    quant = size(elem);
    quant = quant(1);

    if (quant <= 270)
        im_out(elem) = 0;
    end
end

% APLICANDO A OPERACAO MORFOLOGICA DE FECHAMENTO
bw = imerode(im_out, ovo);
bw2 = imdilate(bw, ovo);
im_out = bw2;

% CONTANDO AS AREAS EM BRANCO E DIVIDINDO PELO TAMANHO DO OVO MEDIO
quant2 = find (bw2 == 1);
quant2 = size (quant2);
total = round(quant2(1)/270);
end

```

## Anexo 4

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% UNIVERSIDADE FEDERAL DE PERNAMBUCO %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% CENTRO DE INFORMATICA %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% DIOGENES DOS SANTOS SILVA %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% 14/07/2017 %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% METODO 2 COM MODELO DE CORES YIQ E THRESHOLD FIXO %%%

function [total] = YIQ_Threshold(imagem, ovo)

% LEITURA DA IMAGEM
rgb = imread(imagem);
[lin, col, plan] = size(rgb);

% COLOCANDO A IMAGEM EM FORMATO RGB, CASO NAO ESTEJA
if(size(rgb,3) == 1)
    rgb = cat(3, rgb, rgb, rgb);
end

% CONVERTENDO A IMAGEM PARA O MODELO DE CORES YIQ
% FUNCAO COLORSPACE CRIADA POR PASCAL GETREUER
% DISPONIVEL EM:
% https://www.mathworks.com/matlabcentral/fileexchange/28790-colorspace-transformations?requestedDomain=www.mathworks.com
yiq = colorspace ('RGB->YIQ', rgb);

% CAPTURANDO O COMPONENTE I E SUAS DIMENSOES
band_i = yiq(:,:,2);

```

```

[lin, col, plan] = size(band_i);

% APLICANDO THRESHOLD FIXO DE 200 E MODIFICANDO OS PIXELS PARA:
t = 0.002;
im = band_i;

for i=1:lin
    for j=1:col
        if(im(i,j) > t)
            im(i,j) = 0;
        else
            im(i,j) = 255;
        end
    end
end

% APLICANDO O ALGORITMO DE COMPONENTES CONEXOS
[L, num] = bwlabel (im);
im_out = im;

% REMOVENDO AREAS INFERIORES A 270 PIXELS
for i=1:num
    elem = find(L == i);
    quant = size(elem);
    quant = quant(1);

    if (quant <= 270)
        im_out(elem) = 0;
    end
end

% APLICANDO A OPERACAO MORFOLOGICA DE FECHAMENTO
bw = imerode(im_out, ovo);
bw2 = imdilate(bw, ovo);
im_out = bw2;

% CONTANDO AS AREAS EM BRANCO E DIVIDINDO PELO TAMANHO DO OVO MEDIO
quant2 = find (bw2 == 255);
quant2 = size (quant2);
total = round(quant2(1)/220);
end

```

## Anexo 5

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% UNIVERSIDADE FEDERAL DE PERNAMBUCO %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% CENTRO DE INFORMATICA %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% DIOGENES DOS SANTOS SILVA %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% 14/07/2017 %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% METODO 3 COM MODELO DE CORES LAB E THRESHOLD FIXO %%%

function [total] = LAB(imagem, ovo)

% LEITURA DA IMAGEM
rgb = imread(imagem);
[lin, col, plan] = size(rgb);

% COLOCANDO A IMAGEM EM FORMATO RGB, CASO NAO ESTEJA
if(size(rgb,3) == 1)
    rgb = cat(3, rgb, rgb, rgb);
end

```



```

end

% CONVERTENDO A IMAGEM PARA O MODELO DE CORES LAB
% FUNCAO COLORSPACE CRIADA POR PASCAL GETREUER
% DISPONIVEL EM:
% https://www.mathworks.com/matlabcentral/fileexchange/28790-colorspace-
transformations?requestedDomain=www.mathworks.com
im = colorspace ('RGB->Lab', rgb);

% APLICANDO THRESHOLD FIXO
tmp = uint8(im(:,:,3));
th = 0;
tmp = double(tmp);
for i=1:lin
    for j=1:col
        if (tmp(i,j) > th)
            bw(i,j) = 0;
        else
            bw(i,j) = 255;
        end
    end
end
end

bw = uint8(bw);
im = bw;

% APLICANDO O ALGORITMO DE COMPONENTES CONEXOS
[L, num] = bwlabel (im);
im_out = im;

% REMOVENDO AREAS INFERIORES A 270 PIXELS
for i=1:num
    elem = find(L == i);
    quant = size(elem);
    quant = quant(1);

    if (quant <= 270)
        im_out(elem) = 0;
    end
end
end

% APLICANDO A OPERACAO MORFOLOGICA DE FECHAMENTO
bw = imerode(im_out, ovo);
bw2 = imdilate(bw, ovo);

% CONTANDO AS AREAS EM BRANCO E DIVIDINDO PELO TAMANHO DO OVO MEDIO
quant2 = find (bw2 == 0);
quant2 = size(quant2);
quant2 = quant2(1);
quant2 = lin*col - quant2;
total = round(quant2(1)/270);
end

```

## Anexo 6

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% UNIVERSIDADE FEDERAL DE PERNAMBUCO %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% CENTRO DE INFORMATICA %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% DIOGENES DOS SANTOS SILVA %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% 14/07/2017 %%%%%%%%%

```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% MAIN ROTACAO DE OVOS %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
% LEITURA DAS IMAGENS
```

```
semovos = 'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\sem_ovos.bmp';  
ovos3 = 'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\3ovos.bmp';  
ovos8 = 'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\8ovos.bmp';  
ovos19 = 'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\19ovos.bmp';  
ovos22 = 'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\22ovos.bmp';  
ovos30 = 'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\30ovos.bmp';  
ovos34 = 'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\34ovos.bmp';  
ovos111 = 'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\111ovos.bmp';
```

```
% LEITURA DO ELEMENTO ESTRUTURANTE
```

```
ovosell =  
imread('C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\ovo_sel_18x30.bmp'  
');
```

```
% CRIANDO UM ARRAY DE ARRAYS PARA ARMAZENAR OS CALCULOS DOS OVOS  
ROTACIONADOS
```

```
% PARA CADA METODO DE CONTAGEM
```

```
O = cell(1, 7);  
O(:) = {zeros(1)};  
O3 = cell(1, 7);  
O3(:) = {zeros(1)};  
O8 = cell(1, 7);  
O8(:) = {zeros(1)};  
O19 = cell(1, 7);  
O19(:) = {zeros(1)};  
O22 = cell(1, 7);  
O22(:) = {zeros(1)};  
O30 = cell(1, 7);  
O30(:) = {zeros(1)};  
O34 = cell(1, 7);  
O34(:) = {zeros(1)};  
O111 = cell(1, 7);  
O111(:) = {zeros(1)};
```

```
cont = [1, 1, 1, 1, 1, 1, 1];
```

```
% VARIANDO OS ANGULOS ENTRE 1 E 360 GRAUS
```

```
for i=1:360
```

```
    % VARIANDO O ANGULO A CADA 1 GRAU
```

```
    if(mod(i,1)==0)
```

```
        O{1}(cont(1)) = HSL_Otsu(semovos, imrotate(ovosell, i));  
        O3{1}(cont(1)) = HSL_Otsu(ovos3, imrotate(ovosell, i));  
        O8{1}(cont(1)) = HSL_Otsu(ovos8, imrotate(ovosell, i));  
        O19{1}(cont(1)) = HSL_Otsu(ovos19, imrotate(ovosell, i));  
        O22{1}(cont(1)) = HSL_Otsu(ovos22, imrotate(ovosell, i));  
        O30{1}(cont(1)) = HSL_Otsu(ovos30, imrotate(ovosell, i));  
        O34{1}(cont(1)) = HSL_Otsu(ovos34, imrotate(ovosell, i));  
        O111{1}(cont(1)) = HSL_Otsu(ovos111, imrotate(ovosell, i));  
        cont(1) = cont(1) + 1;
```

```
    end
```

```
    % VARIANDO O ANGULO A CADA 2 GRAUS
```

```
    if(mod(i,2)==0)
```

```
        O{2}(cont(2)) = HSL_Otsu(semovos, imrotate(ovosell, i));  
        O3{2}(cont(2)) = HSL_Otsu(ovos3, imrotate(ovosell, i));  
        O8{2}(cont(2)) = HSL_Otsu(ovos8, imrotate(ovosell, i));  
        O19{2}(cont(2)) = HSL_Otsu(ovos19, imrotate(ovosell, i));  
        O22{2}(cont(2)) = HSL_Otsu(ovos22, imrotate(ovosell, i));
```

```

    O30{2}(cont(2)) = HSL_Otsu(ovos30, imrotate(ovosell, i));
    O34{2}(cont(2)) = HSL_Otsu(ovos34, imrotate(ovosell, i));
    O111{2}(cont(2)) = HSL_Otsu(ovos111, imrotate(ovosell, i));
    cont(2) = cont(2) + 1;
end
% VARIANDO O ANGULO A CADA 3 GRAUS
if(mod(i,3)==0)
    O{3}(cont(3)) = HSL_Otsu(semovos, imrotate(ovosell, i));
    O3{3}(cont(3)) = HSL_Otsu(ovos3, imrotate(ovosell, i));
    O8{3}(cont(3)) = HSL_Otsu(ovos8, imrotate(ovosell, i));
    O19{3}(cont(3)) = HSL_Otsu(ovos19, imrotate(ovosell, i));
    O22{3}(cont(3)) = HSL_Otsu(ovos22, imrotate(ovosell, i));
    O30{3}(cont(3)) = HSL_Otsu(ovos30, imrotate(ovosell, i));
    O34{3}(cont(3)) = HSL_Otsu(ovos34, imrotate(ovosell, i));
    O111{3}(cont(3)) = HSL_Otsu(ovos111, imrotate(ovosell, i));
    cont(3) = cont(3) + 1;
end
% VARIANDO O ANGULO A CADA 4 GRAUS
if(mod(i,4)==0)
    O{4}(cont(4)) = HSL_Otsu(semovos, imrotate(ovosell, i));
    O3{4}(cont(4)) = HSL_Otsu(ovos3, imrotate(ovosell, i));
    O8{4}(cont(4)) = HSL_Otsu(ovos8, imrotate(ovosell, i));
    O19{4}(cont(4)) = HSL_Otsu(ovos19, imrotate(ovosell, i));
    O22{4}(cont(4)) = HSL_Otsu(ovos22, imrotate(ovosell, i));
    O30{4}(cont(4)) = HSL_Otsu(ovos30, imrotate(ovosell, i));
    O34{4}(cont(4)) = HSL_Otsu(ovos34, imrotate(ovosell, i));
    O111{4}(cont(4)) = HSL_Otsu(ovos111, imrotate(ovosell, i));
    cont(4) = cont(4) + 1;
end
% VARIANDO O ANGULO A CADA 5 GRAUS
if(mod(i,5)==0)
    O{5}(cont(5)) = HSL_Otsu(semovos, imrotate(ovosell, i));
    O3{5}(cont(5)) = HSL_Otsu(ovos3, imrotate(ovosell, i));
    O8{5}(cont(5)) = HSL_Otsu(ovos8, imrotate(ovosell, i));
    O19{5}(cont(5)) = HSL_Otsu(ovos19, imrotate(ovosell, i));
    O22{5}(cont(5)) = HSL_Otsu(ovos22, imrotate(ovosell, i));
    O30{5}(cont(5)) = HSL_Otsu(ovos30, imrotate(ovosell, i));
    O34{5}(cont(5)) = HSL_Otsu(ovos34, imrotate(ovosell, i));
    O111{5}(cont(5)) = HSL_Otsu(ovos111, imrotate(ovosell, i));
    cont(5) = cont(5) + 1;
end
% VARIANDO O ANGULO A CADA 10 GRAUS
if(mod(i,10)==0)
    O{6}(cont(6)) = HSL_Otsu(semovos, imrotate(ovosell, i));
    O3{6}(cont(6)) = HSL_Otsu(ovos3, imrotate(ovosell, i));
    O8{6}(cont(6)) = HSL_Otsu(ovos8, imrotate(ovosell, i));
    O19{6}(cont(6)) = HSL_Otsu(ovos19, imrotate(ovosell, i));
    O22{6}(cont(6)) = HSL_Otsu(ovos22, imrotate(ovosell, i));
    O30{6}(cont(6)) = HSL_Otsu(ovos30, imrotate(ovosell, i));
    O34{6}(cont(6)) = HSL_Otsu(ovos34, imrotate(ovosell, i));
    O111{6}(cont(6)) = HSL_Otsu(ovos111, imrotate(ovosell, i));
    cont(6) = cont(6) + 1;
end
% VARIANDO O ANGULO A CADA 15 GRAUS
if(mod(i,15)==0)
    O{7}(cont(7)) = HSL_Otsu(semovos, imrotate(ovosell, i));
    O3{7}(cont(7)) = HSL_Otsu(ovos3, imrotate(ovosell, i));
    O8{7}(cont(7)) = HSL_Otsu(ovos8, imrotate(ovosell, i));
    O19{7}(cont(7)) = HSL_Otsu(ovos19, imrotate(ovosell, i));
    O22{7}(cont(7)) = HSL_Otsu(ovos22, imrotate(ovosell, i));
    O30{7}(cont(7)) = HSL_Otsu(ovos30, imrotate(ovosell, i));

```

```

O34{7}(cont(7)) = HSL_Otsu(ovos34, imrotate(ovosell, i));
O111{7}(cont(7)) = HSL_Otsu(ovos111, imrotate(ovosell, i));
cont(7) = cont(7) + 1;
end
end

% ESCRREVENDO OS RESULTADOS EM UM ARQUIVO
fileID =
fopen('C:\Users\devil\OneDrive\Documentos\TG\Projeto\Otsu_Rotacao.txt','w')
;
fprintf(fileID,'%s\r\n','##### Rotacao de ovos com o metodo de Otsu
#####');
fprintf(fileID,'%s\r\n','##### sem ovos #####');
fprintf(fileID,'%s      %s \r\n','##### Media','Desvio Padrao
#####');
fprintf(fileID,'%f %s %f\r\n',mean(O{1}),';', std(O{1}));
fprintf(fileID,'%f %s %f\r\n',mean(O{2}),';', std(O{2}));
fprintf(fileID,'%f %s %f\r\n',mean(O{3}),';', std(O{3}));
fprintf(fileID,'%f %s %f\r\n',mean(O{4}),';', std(O{4}));
fprintf(fileID,'%f %s %f\r\n',mean(O{5}),';', std(O{5}));
fprintf(fileID,'%f %s %f\r\n',mean(O{6}),';', std(O{6}));
fprintf(fileID,'%f %s %f\r\n',mean(O{7}),';', std(O{7}));
fprintf(fileID,'%s\r\n','##### 3 ovos #####');
fprintf(fileID,'%s      %s \r\n','##### Media','Desvio Padrao
#####');
fprintf(fileID,'%f %s %f\r\n',mean(O3{1}),';', std(O3{1}));
fprintf(fileID,'%f %s %f\r\n',mean(O3{2}),';', std(O3{2}));
fprintf(fileID,'%f %s %f\r\n',mean(O3{3}),';', std(O3{3}));
fprintf(fileID,'%f %s %f\r\n',mean(O3{4}),';', std(O3{4}));
fprintf(fileID,'%f %s %f\r\n',mean(O3{5}),';', std(O3{5}));
fprintf(fileID,'%f %s %f\r\n',mean(O3{6}),';', std(O3{6}));
fprintf(fileID,'%f %s %f\r\n',mean(O3{7}),';', std(O3{7}));
fprintf(fileID,'%s\r\n','##### 8 ovos #####');
fprintf(fileID,'%s      %s \r\n','##### Media','Desvio Padrao
#####');
fprintf(fileID,'%f %s %f\r\n',mean(O8{1}),';', std(O8{1}));
fprintf(fileID,'%f %s %f\r\n',mean(O8{2}),';', std(O8{2}));
fprintf(fileID,'%f %s %f\r\n',mean(O8{3}),';', std(O8{3}));
fprintf(fileID,'%f %s %f\r\n',mean(O8{4}),';', std(O8{4}));
fprintf(fileID,'%f %s %f\r\n',mean(O8{5}),';', std(O8{5}));
fprintf(fileID,'%f %s %f\r\n',mean(O8{6}),';', std(O8{6}));
fprintf(fileID,'%f %s %f\r\n',mean(O8{7}),';', std(O8{7}));
fprintf(fileID,'%s\r\n','##### 19 ovos #####');
fprintf(fileID,'%s      %s \r\n','##### Media','Desvio Padrao
#####');
fprintf(fileID,'%f %s %f\r\n',mean(O19{1}),';', std(O19{1}));
fprintf(fileID,'%f %s %f\r\n',mean(O19{2}),';', std(O19{2}));
fprintf(fileID,'%f %s %f\r\n',mean(O19{3}),';', std(O19{3}));
fprintf(fileID,'%f %s %f\r\n',mean(O19{4}),';', std(O19{4}));
fprintf(fileID,'%f %s %f\r\n',mean(O19{5}),';', std(O19{5}));
fprintf(fileID,'%f %s %f\r\n',mean(O19{6}),';', std(O19{6}));
fprintf(fileID,'%f %s %f\r\n',mean(O19{7}),';', std(O19{7}));
fprintf(fileID,'%s\r\n','##### 22 ovos #####');
fprintf(fileID,'%s      %s \r\n','##### Media','Desvio Padrao
#####');
fprintf(fileID,'%f %s %f\r\n',mean(O22{1}),';', std(O22{1}));
fprintf(fileID,'%f %s %f\r\n',mean(O22{2}),';', std(O22{2}));
fprintf(fileID,'%f %s %f\r\n',mean(O22{3}),';', std(O22{3}));
fprintf(fileID,'%f %s %f\r\n',mean(O22{4}),';', std(O22{4}));
fprintf(fileID,'%f %s %f\r\n',mean(O22{5}),';', std(O22{5}));
fprintf(fileID,'%f %s %f\r\n',mean(O22{6}),';', std(O22{6}));

```

```

fprintf(fileID, '%f %s %f\r\n', mean(O22{7}), ';', std(O22{7}));
fprintf(fileID, '%s\r\n', '##### 30 ovos #####');
fprintf(fileID, '%s      %s \r\n', '##### Media', 'Desvio Padrao
#####');
fprintf(fileID, '%f %s %f\r\n', mean(O30{1}), ';', std(O30{1}));
fprintf(fileID, '%f %s %f\r\n', mean(O30{2}), ';', std(O30{2}));
fprintf(fileID, '%f %s %f\r\n', mean(O30{3}), ';', std(O30{3}));
fprintf(fileID, '%f %s %f\r\n', mean(O30{4}), ';', std(O30{4}));
fprintf(fileID, '%f %s %f\r\n', mean(O30{5}), ';', std(O30{5}));
fprintf(fileID, '%f %s %f\r\n', mean(O30{6}), ';', std(O30{6}));
fprintf(fileID, '%f %s %f\r\n', mean(O30{7}), ';', std(O30{7}));
fprintf(fileID, '%s\r\n', '##### 34 ovos #####');
fprintf(fileID, '%s      %s \r\n', '##### Media', 'Desvio Padrao
#####');
fprintf(fileID, '%f %s %f\r\n', mean(O34{1}), ';', std(O34{1}));
fprintf(fileID, '%f %s %f\r\n', mean(O34{2}), ';', std(O34{2}));
fprintf(fileID, '%f %s %f\r\n', mean(O34{3}), ';', std(O34{3}));
fprintf(fileID, '%f %s %f\r\n', mean(O34{4}), ';', std(O34{4}));
fprintf(fileID, '%f %s %f\r\n', mean(O34{5}), ';', std(O34{5}));
fprintf(fileID, '%f %s %f\r\n', mean(O34{6}), ';', std(O34{6}));
fprintf(fileID, '%f %s %f\r\n', mean(O34{7}), ';', std(O34{7}));
fprintf(fileID, '%s\r\n', '##### 111 ovos #####');
fprintf(fileID, '%s      %s \r\n', '##### Media', 'Desvio Padrao
#####');
fprintf(fileID, '%f %s %f\r\n', mean(O111{1}), ';', std(O111{1}));
fprintf(fileID, '%f %s %f\r\n', mean(O111{2}), ';', std(O111{2}));
fprintf(fileID, '%f %s %f\r\n', mean(O111{3}), ';', std(O111{3}));
fprintf(fileID, '%f %s %f\r\n', mean(O111{4}), ';', std(O111{4}));
fprintf(fileID, '%f %s %f\r\n', mean(O111{5}), ';', std(O111{5}));
fprintf(fileID, '%f %s %f\r\n', mean(O111{6}), ';', std(O111{6}));
fprintf(fileID, '%f %s %f\r\n', mean(O111{7}), ';', std(O111{7}));
fclose(fileID);

```

## Anexo 7

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% UNIVERSIDADE FEDERAL DE PERNAMBUCO %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% CENTRO DE INFORMATICA %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% DIOGENES DOS SANTOS SILVA %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% 14/07/2017 %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% METODO QUE DIVIDE AS IMAGENS %%%%%%%%%

% IMAGENS
ovos34 = 'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\34ovos.bmp';
ovos111 = 'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\111ovos.bmp';

% LEITURA DO ELEMENTO ESTRUTURANTE
ovosell =
imread('C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\ovo_sel_18x30.bmp
');

% LEITURA DA IMAGEM
im = imread(ovos34);
[lin, col, plan] = size(im);

% CALCULO DE OVOS PARA CADA K DIVISOES
for k=1:10
    divisor = k;

    % CAPTURANDO O NUMERO DIVISOES PARA A LINHA E A COLUNA

```

```

div_lin = ceil(lin/divisor);
div_col = ceil(col/divisor);

% ARMAZENANDO ESPACO PARA CADA ELEMENTO DA MATRIZ
im_cropped = cell(1, divisor^2);

% RECORTANDO A IMAGEM
cont = 0;
for i=1:divisor
    for j=1:divisor
        cont = cont + 1;
        im_cropped{cont} = imcrop(im, [((i-1)*div_col) ((j-1)*div_lin)
div_col div_lin]);
    end
end

total_yiq = 0;
total_lab = 0;
total_otsu = 0;
total_kittler = 0;

% CONTAGEM DE OVOS PARA CADA PARTE RECORTADA
for i=1:(divisor^2)
    if(isempty(im_cropped{i}) == 0)
        total_yiq = total_yiq + YIQ_Threshold(im_cropped{i}, ovosell);
        total_lab = total_lab + LAB(im_cropped{i}, ovosell);
        total_otsu = total_otsu + HSV_Otsu(im_cropped{i}, ovosell);
        total_kittler = total_kittler + HSV_Kittler(im_cropped{i},
ovosell);
    end
end

% MOSTRANDO O VALOR TOTAL DE OVOS RETORNADO
display(['##### Contagem de ovos com ' int2str(k) '
divisoess #####']);
display(['Contagem com YIQ = ' int2str(total_yiq)]);
display(['Contagem com LAB = ' int2str(total_lab)]);
display(['Contagem com OTSU = ' int2str(total_otsu)]);
display(['Contagem com KITTLER = ' int2str(total_kittler)]);
display('#####');

vars = {'div_lin','div_col','im_cropped', 'total_yiq', 'total_lab'};
clear (vars{:});
end

```

## Anexo 8

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% UNIVERSIDADE FEDERAL DE PERNAMBUCO %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% CENTRO DE INFORMATICA %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% DIOGENES DOS SANTOS SILVA %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% 14/07/2017 %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% MAIN PARA IMAGENS COM AMPLIACAO %%%%%%%%%

% IMAGENS
nova4 = 'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img_novas\Zoom
Maior\nova4.jpg';
nova5 = 'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img_novas\Zoom
Maior\nova5.jpg';
nova6 = 'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img_novas\Zoom
Maior\nova6.jpg';

```

```

nova7 = 'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img_novas\Zoom
Maior\nova7.jpg';
nova8 = 'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img_novas\Zoom
Maior\nova8.jpg';
nova9 = 'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img_novas\Zoom
Maior\nova9.jpg';

% LEITURA DA IMAGEM
im = imread(nova4);
[lin, col, plan] = size(im);

% QUANTIDADE DE OVOS RETORNADA PELOS METODOS 1, 2 E 3
total_otstu = HSL_Otsu(im);
total_kittler = HSL_Kittler(im);
total_yiq = YIQ_Threshold(im);
total_lab = LAB(im);

% MOSTRANDO A QUANTIDADE ENCONTRADA
display(['otsu: ' int2str(total_otstu)]);
display(['kittler: ' int2str(total_kittler)]);
display(['yiq: ' int2str(total_yiq)]);
display(['lab: ' int2str(total_lab)]);

vars = {'ovosel', 'ovosell', 'im', 'col', 'lin', 'plan'};
clear (vars{:});

```

## Anexo 9

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% UNIVERSIDADE FEDERAL DE PERNAMBUCO %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% CENTRO DE INFORMATICA %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% DIOGENES DOS SANTOS SILVA %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% 14/07/2017 %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% METODO 1 COM A BINARIZACAO DE OTSU %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% PARA IMAGENS COM AMPLIACAO %%%%%%%%%

function [total] = HSL_Otsu(imagem)

% LEITURA DA IMAGEM
rgb = imagem;

% AJUSTES DA IMAGEM
rgb = imadjust(rgb, [.135 0 0; 1 1 1], stretchlim(rgb, 0.5), 0.1);

% COLOCANDO A IMAGEM EM FORMATO RGB, CASO NAO ESTEJA
if(size(rgb,3) == 1)
    rgb = cat(3, rgb, rgb, rgb);
end

% LEITURA DO ELEMENTO ESTRUTURANTE
ovosel =
imread('C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\ovo_sel_8x13.bmp'
);

% CONVERTENDO A IMAGEM PARA O MODELO DE CORES HSV
rgb = rgb2hsv(rgb);

% APLICANDO A BINARIZACAO DE OTSU
level = graythresh(rgb(:,:,1));
rgb = rgb(:,:,1) > level;

```

```

% APLICANDO O ALGORITMO DE COMPONENTES CONEXOS
[L, num] = bwlabel (rgb(:,:,1));
im_out = rgb(:,:,1);

% REMOVENDO AREAS INFERIORES A 430 PIXELS
for i=1:num
    elem = find(L == i);
    quant = size(elem);
    quant = quant(1);

    if (quant <= 430)
        im_out(elem) = 0;
    end
end

% APLICANDO A OPERACAO MORFOLOGICA DE DILATACAO
bw2 = imdilate(im_out, ovosel);
im_out = bw2;

% CONTANDO AS AREAS EM BRANCO E DIVIDINDO PELO TAMANHO DO OVO MEDIO
quant2 = find (im_out == 1);
quant2 = size (quant2);
total = round(quant2(1)/2200);
end

```

## Anexo 10

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% UNIVERSIDADE FEDERAL DE PERNAMBUCO %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% CENTRO DE INFORMATICA %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% DIOGENES DOS SANTOS SILVA %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% 14/07/2017 %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% METODO 1 COM A BINARIZACAO DE KITTLER %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% PARA IMAGENS COM AMPLIACAO %%%%%%%%%

function [total] = HSL_Kittler(imagem)

% LEITURA DA IMAGEM
rgb = imagem;

% AJUSTES DA IMAGEM
rgb = imadjust(rgb,[.145 0 0; 1 1 1],stretchlim(rgb, 0.5), 0.05);

% COLOCANDO A IMAGEM EM FORMATO RGB, CASO NAO ESTEJA
if(size(rgb,3) == 1)
    rgb = cat(3, rgb, rgb, rgb);
end

% LEITURA DO ELEMENTO ESTRUTURANTE
ovosel =
imread('C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\ovo_sel_8x13.bmp'
);

% CONVERTENDO A IMAGEM PARA O MODELO DE CORES HSV
rgb = rgb2hsv(rgb);

% APLICANDO A BINARIZACAO DE KITTLER
level = Kittler(rgb(:,:,1));
rgb = rgb(:,:,1) > level;

```



```

% APLICANDO O ALGORITMO DE COMPONENTES CONEXOS
[L, num] = bwlabel (rgb(:,:,1));
im_out = rgb(:,:,1);

% REMOVENDO AREAS INFERIORES A 430 PIXELS
for i=1:num
    elem = find(L == i);
    quant = size(elem);
    quant = quant(1);

    if (quant <= 430)
        im_out(elem) = 0;
    end
end

% APLICANDO A OPERACAO MORFOLOGICA DE DILATACAO
bw2 = imdilate(im_out, ovosel);
im_out = bw2;

% CONTANDO AS AREAS EM BRANCO E DIVIDINDO PELO TAMANHO DO OVO MEDIO
quant2 = find (im_out == 1);
quant2 = size (quant2);
total = round(quant2(1)/2200);
end

```

## Anexo 11

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% UNIVERSIDADE FEDERAL DE PERNAMBUCO %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% CENTRO DE INFORMATICA %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% DIOGENES DOS SANTOS SILVA %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% 14/07/2017 %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% METODO 3 COM MODELO DE CORES LAB %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% E THRESHOLD FIXO PARA IMAGENS %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% COM AMPLIACAO %%%%%%%%%

function [total] = LAB (imagem)

% LEITURA DA IMAGEM
im2 = imagem;

% AJUSTES DA IMAGEM
im2 = imadjust(im2,[.135 0.135 0; 1 .6 1], stretchlim(im2, 0.5), 1);
[lin, col, plan] = size(im2);

% COLOCANDO A IMAGEM EM FORMATO RGB, CASO NAO ESTEJA
if(size(im2,3) == 1)
    im2 = cat(3, im2, im2, im2);
end

% LEITURA DO ELEMENTO ESTRUTURANTE
ovosel =
imread('C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\ovo_sel_8x13.bmp'
);

% CONVERTENDO A IMAGEM PARA O MODELO DE CORES LAB
im = colorspace ('RGB->Lab', im2);
tmp = uint8(im(:,:,3));

```

```

% APLICANDO THRESHOLD FIXO
th = 0;
tmp = double(tmp);
for i=1:lin
    for j=1:col
        if (tmp(i,j) > th)
            bw(i,j) = 0;
        else
            bw(i,j) = 255;
        end
    end
end

bw = uint8(bw);
im = bw;

% APLICANDO O ALGORITMO DE COMPONENTES CONEXOS
[L, num] = bwlabel (im);
im_out = im;

% REMOVENDO AREAS INFERIORES A 430 PIXELS
for i=1:num
    elem = find(L == i);
    quant = size(elem);
    quant = quant(1);
    if (quant <= 430)
        im_out (elem) = 0;
    end
end

% APLICANDO A OPERACAO MORFOLOGICA DE DILATACAO
bw2 = imdilate(im_out, ovesel);
im_out = bw2;

% CONTANDO AS AREAS EM BRANCO E DIVIDINDO PELO TAMANHO DO OVO MEDIO
quant2 = find (im_out == 0);
quant2 = size(quant2);
quant2 = quant2(1);
quant2 = lin*col - quant2;
total = round(quant2/2200);
end

```

## Anexo 12

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% UNIVERSIDADE FEDERAL DE PERNAMBUCO %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% CENTRO DE INFORMATICA %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% DIOGENES DOS SANTOS SILVA %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% 14/07/2017 %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% METODO 2 COM MODELO DE CORES YIQ %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% E THRESHOLD FIXO PARA IMAGENS %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% COM AMPLIACAO %%%%%%%%%

```

```

function [total] = YIQ_Threshold(imagem)

% LEITURA DA IMAGEM
rgb = imagem;

% AJUSTES DA IMAGEM
rgb = imadjust(rgb, [.11 0 0; .9 .95 1], stretchlim(rgb, 0.5), 0.5);

```

```

% COLOCANDO A IMAGEM EM FORMATO RGB, CASO NAO ESTEJA
if(size(rgb,3) == 1)
    rgb = cat(3, rgb, rgb, rgb);
end

%Leitura da imagem com os ovos de referencia
ovosel =
imread('C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\ovo_sel_8x13.bmp'
);

% CONVERTENDO A IMAGEM PARA O MODELO DE CORES YIQ
yiq = colorspace ('RGB->YIQ', rgb);

% CAPTURANDO O COMPONENTE I
band_i = yiq(:,:,2);
[lin, col, plan] = size(band_i);

% APLICANDO A BINARIZACAO DO THRESHOLD FIXO
t = 0.002;
im = band_i;

for i=1:lin
    for j=1:col
        if(im(i,j) > t)
            im(i,j) = 0;
        else
            im(i,j) = 255;
        end
    end
end

% APLICANDO O ALGORITMO DE COMPONENTES CONEXOS
[L, num] = bwlabel (im);
im_out = im;

% REMOVENDO AREAS INFERIORES A 430 PIXELS
for i=1:num
    elem = find(L == i);
    quant = size(elem);
    quant = quant(1);

    if (quant <= 430)
        im_out(elem) = 0;
    end
end

% APLICANDO A OPERACAO MORFOLOGICA DE DILATACAO
bw2 = imdilate(im_out, ovosel);
im_out = bw2;

% CONTANDO AS AREAS EM BRANCO E DIVIDINDO PELO TAMANHO DO OVO MEDIO
quant2 = find (im_out == 255);
quant2 = size (quant2);
total = round(quant2(1)/2200);
end

```

## Anexo 13

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% UNIVERSIDADE FEDERAL DE PERNAMBUCO %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% CENTRO DE INFORMATICA %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% DIOGENES DOS SANTOS SILVA %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% 14/07/2017 %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% MAIN PARA IMAGENS SEM AMPLIACAO %%%%%%%%%

```

```

% IMAGENS
nova10 =
'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img_novas\nova10.jpg';
nova11 =
'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img_novas\nova11.jpg';
nova12 =
'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img_novas\nova12.jpg';
nova13 =
'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img_novas\nova13.jpg';
nova14 =
'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img_novas\nova14.jpg';
nova15 =
'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img_novas\nova15.jpg';
nova16 =
'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img_novas\nova16.jpg';
nova17 =
'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img_novas\nova17.jpg';
nova18 =
'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img_novas\nova18.jpg';
nova19 =
'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img_novas\nova19.jpg';
nova20 =
'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img_novas\nova20.jpg';
nova21 =
'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img_novas\nova21.jpg';
nova22 =
'C:\Users\devil\OneDrive\Documentos\TG\Projeto\img_novas\nova22.jpg';

```

```

% LEITURA DA IMAGEM
im = imread(nova22);
[lin, col, plan] = size(im);

```

```

% QUANTIDADE DE OVOS RETORNADA PELOS METODOS
total_otstu = HSL_Otsu(im);
total_kittler = HSL_Kittler(im);
total_yiq = YIQ_Threshold(im);
total_lab = LAB(im);

```

```

% MOSTRANDO A QUANTIDADE ENCONTRADA
display(['otsu: ' int2str(total_otstu)]);
display(['kittler: ' int2str(total_kittler)]);
display(['yiq: ' int2str(total_yiq)]);
display(['lab: ' int2str(total_lab)]);
display(['#####']);

```

```

vars = {'ovosel', 'ovosell', 'im', 'col', 'lin', 'plan'};
clear (vars{:});

```

## Anexo 14

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% UNIVERSIDADE FEDERAL DE PERNAMBUCO %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% CENTRO DE INFORMATICA %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% DIOGENES DOS SANTOS SILVA %%%%%%%%%

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% 14/07/2017 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% METODO 1 COM A BINARIZACAO DE OTSU %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% PARA IMAGENS SEM AMPLIACAO %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

function [total] = HSL_Otsu(imagem)

% LEITURA DA IMAGEM
rgb = imagem;

% AJUSTES DA IMAGEM
rgb = imadjust(rgb,[.135 0 0; 1 1 1],stretchlim(rgb, 0.5), 0.1);

% COLOCANDO A IMAGEM EM FORMATO RGB, CASO NAO ESTEJA
if(size(rgb,3) == 1)
    rgb = cat(3, rgb, rgb, rgb);
end

% LEITURA DO ELEMENTO ESTRUTURANTE
ovosel =
imread('C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\ovo_sel_8x13.bmp'
);

% CONVERTENDO A IMAGEM PARA O MODELO DE CORES HSV
rgb = rgb2hsv(rgb);

% APLICANDO A BINARIZACAO DE OTSU
level = graythresh(rgb(:,:,1));
rgb = rgb(:,:,1) > level;

% APLICANDO O ALGORITMO DE COMPONENTES CONEXOS
[L, num] = bwlabel (rgb(:,:,1));
im_out = rgb(:,:,1);

% REMOVENDO AREAS INFERIORES A 320 PIXELS
for i=1:num
    elem = find(L == i);
    quant = size(elem);
    quant = quant(1);

    if (quant <= 320)
        im_out(elem) = 0;
    end
end

% APLICANDO A OPERACAO MORFOLOGICA DE DILATAcao
bw2 = imdilate(im_out, ovosel);
im_out = bw2;

% CONTANDO AS AREAS EM BRANCO E DIVIDINDO PELO TAMANHO DO OVO MEDIO
quant2 = find (im_out == 1);
quant2 = size (quant2);
total = round(quant2(1)/1475);
end

```

## Anexo 15

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% UNIVERSIDADE FEDERAL DE PERNAMBUCO %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% CENTRO DE INFORMATICA %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% DIOGENES DOS SANTOS SILVA %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% 14/07/2017 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% METODO 1 COM A BINARIZACAO DE KITTLER %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% PARA IMAGENS SEM AMPLIACAO %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

function [total] = HSL_Kittler(imagem)

% LEITURA DA IMAGEM
rgb = imagem;

% AJUSTES DA IMAGEM
rgb = imadjust(rgb,[.145 0 0; 1 1 1],stretchlim(rgb, 0.5), 0.05);

% COLOCANDO A IMAGEM EM FORMATO RGB, CASO NAO ESTEJA
if(size(rgb,3) == 1)
    rgb = cat(3, rgb, rgb, rgb);
end

% LEITURA DO ELEMENTO ESTRUTURANTE
ovosel1 =
imread('C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\ovoselnew.bmp');

% CONVERTENDO A IMAGEM PARA O MODELO DE CORES HSV
rgb = rgb2hsv(rgb);

% APLICANDO A BINARIZACAO DE KITTLER
level = Kittler(rgb(:,:,1));
rgb = rgb(:,:,1) > level;

% APLICANDO O ALGORITMO DE COMPONENTES CONEXOS
[L, num] = bwlabel (rgb(:,:,1));
im_out = rgb(:,:,1);

% REMOVENDO AREAS INFERIORES A 320 PIXELS
for i=1:num
    elem = find(L == i);
    quant = size(elem);
    quant = quant(1);

    if (quant <= 320)
        im_out(elem) = 0;
    end
end

% APLICANDO A OPERACAO MORFOLOGICA DE DILATAcao
bw2 = imdilate(im_out, ovosel1);
im_out = bw2;

% CONTANDO AS AREAS EM BRANCO E DIVIDINDO PELO TAMANHO DO OVO MEDIO
quant2 = find (im_out == 1);
quant2 = size (quant2);
total = round(quant2(1)/1475);
end

```

## Anexo 16

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% UNIVERSIDADE FEDERAL DE PERNAMBUCO %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% CENTRO DE INFORMATICA %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% DIOGENES DOS SANTOS SILVA %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% 14/07/2017 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% METODO 2 COM MODELO DE CORES YIQ %%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% E THRESHOLD FIXO PARA IMAGENS %%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% SEM AMPLIACAO %%%%%%%%%%

function [total] = YIQ_Threshold(imagem)

% LEITURA DA IMAGEM
rgb = imagem;

% AJUSTES DA IMAGEM
rgb = imadjust(rgb, [.11 0 0; .9 .95 1],stretchlim(rgb, 0.5), 0.5);

% COLOCANDO A IMAGEM EM FORMATO RGB, CASO NAO ESTEJA
if(size(rgb,3) == 1)
    rgb = cat(3, rgb, rgb, rgb);
end

%Leitura da imagem com os ovos de referencia
ovosel =
imread('C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\ovo_sel_8x13.bmp'
);

% CONVERTENDO A IMAGEM PARA O MODELO DE CORES YIQ
yiq = colorspace ('RGB->YIQ', rgb);

% CAPTURANDO O COMPONENTE I
band_i = yiq(:,:,2);
[lin, col, plan] = size(band_i);

% APLICANDO A BINARIZACAO DO THRESHOLD FIXO
t = 0.002;
im = band_i;

for i=1:lin
    for j=1:col
        if(im(i,j) > t)
            im(i,j) = 0;
        else
            im(i,j) = 255;
        end
    end
end

% APLICANDO O ALGORITMO DE COMPONENTES CONEXOS
[L, num] = bwlabel (im);
im_out = im;

% REMOVENDO AREAS INFERIORES A 320 PIXELS
for i=1:num
    elem = find(L == i);
    quant = size(elem);
    quant = quant(1);

    if (quant <= 320)
        im_out(elem) = 0;
    end
end

% APLICANDO A OPERACAO MORFOLOGICA DE DILATACAO
bw2 = imdilate(im_out, ovosel);
im_out = bw2;

```

```

% CONTANDO AS AREAS EM BRANCO E DIVIDINDO PELO TAMANHO DO OVO MEDIO
quant2 = find (im_out == 255);
quant2 = size (quant2);
total = round(quant2(1)/1475);
end

```

## Anexo 17

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
UNIVERSIDADE FEDERAL DE PERNAMBUCO
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
CENTRO DE INFORMATICA
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
DIOGENES DOS SANTOS SILVA
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
14/07/2017
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
METODO 3 COM MODELO DE CORES LAB
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
E THRESHOLD FIXO PARA IMAGENS
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
SEM AMPLIACAO
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

function [total] = LAB (imagem)

% LEITURA DA IMAGEM
im2 = imagem;

% AJUSTES DA IMAGEM
im2 = imadjust(im2,[.135 0.135 0; 1 .6 1], stretchlim(im2, 0.5), 1);
[lin, col, plan] = size(im2);

% COLOCANDO A IMAGEM EM FORMATO RGB, CASO NAO ESTEJA
if(size(im2,3) == 1)
    im2 = cat(3, im2, im2, im2);
end

%Leitura da imagem com os ovos de referencia
ovosel =
imread('C:\Users\devil\OneDrive\Documentos\TG\Projeto\img\ovo_sel_8x13.bmp'
);

% CONVERTENDO A IMAGEM PARA O MODELO DE CORES LAB
im = colorspace ('RGB->Lab', im2);
tmp = uint8(im(:,:,3));

% APLICANDO THRESHOLD FIXO
th = 0;
tmp = double(tmp);
for i=1:lin
    for j=1:col
        if (tmp(i,j) > th)
            bw(i,j) = 0;
        else
            bw(i,j) = 255;
        end
    end
end
end

bw = uint8(bw);
im = bw;

% APLICANDO O ALGORITMO DE COMPONENTES CONEXOS
[L, num] = bwlabel (im);
im_out = im;

```



```

% REMOVENDO AREAS INFERIORES A 320 PIXELS
for i=1:num
    elem = find(L == i);
    quant = size(elem);
    quant = quant(1);
    if (quant <= 320)
        im_out (elem) = 0;
    end
end

% APLICANDO A OPERACAO MORFOLOGICA DE DILATACAO
bw2 = imdilate(im_out, ovosel);
im_out = bw2;

% CONTANDO AS AREAS EM BRANCO E DIVIDINDO PELO TAMANHO DO OVO MEDIO
quant2 = find (im_out == 0);
quant2 = size(quant2);
quant2 = quant2(1);
quant2 = lin*col - quant2;
total = round(quant2/1475);
end

```