

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ  
DEPARTAMENTO ACADÊMICO DE COMPUTAÇÃO  
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE  
SISTEMAS

CLEIFAI A. G. BERLANDA

**CONTAGEM DE AVES DE CORTE EM UM AVIÁRIO ATRAVÉS DO  
PROCESSAMENTO DIGITAL DE IMAGENS**

TRABALHO DE DIPLOMAÇÃO

MEDIANEIRA

2015

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ – UTFPR  
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE  
SISTEMAS

CLEIFAI A. G. BERLANDA

**CONTAGEM DE AVES DE CORTE EM UM AVIÁRIO ATRAVÉS DO  
PROCESSAMENTO DIGITAL DE IMAGENS**

Trabalho de conclusão de curso, apresentado a disciplina de Trabalho de Diplomação, do Curso, Superior de Tecnologia em Análise e Desenvolvimento de Sistemas – COADS – da Universidade Tecnológica Federal do Paraná – UTFPR, como requisito parcial para obtenção do título de Tecnólogo.

Orientador: Prof. Dr. Pedro Luiz de Paula Filho

Co-orientador: Paulo Giovanni de Abreu

MEDIANEIRA

2015

## **DEDICATORIA**

Dedico este trabalho a meus pais Tito Daniel Berlanda, Eliud G. S. Berlanda e a minha Esposa Mariellen S. G. Berlanda, pela paciência e apoio durante a graduação.

Ministério da Educação



**Universidade Tecnológica Federal do Paraná**

Diretoria de Graduação e Educação Profissional

Coordenação do Curso Superior de Tecnologia em  
Análise e Desenvolvimento de Sistemas



---

---

## TERMO DE APROVAÇÃO

### CONTAGEM DE AVES DE CORTE EM UM AVIÁRIO ATRAVÉS DO PROCESSAMENTO DIGITAL DE IMAGENS

Por

**Cleifai A. G. Berlanda**

Este Trabalho de Diplomação (TD) foi apresentado às 14:40 h do dia 12 de Junho de 2015 como requisito parcial para a obtenção do título de Tecnólogo no Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas, da Universidade Tecnológica Federal do Paraná, *Campus* Medianeira. O candidato foi argüido pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

---

Prof. Pedro Luiz de Paula Filho

UTFPR – *Campus* Medianeira

(Orientador)

---

Prof. Arnaldo Cândido Júnior

UTFPR – *Campus* Medianeira

(Convidado)

---

Prof. Jorge Aikes Júnior

UTFPR – *Campus* Medianeira

(Convidado)

---

Prof. Juliano Rodrigo Lamb

UTFPR – *Campus* Medianeira

(Responsável pelas atividades de TCC)

## **AGRADECIMENTOS**

Primeiramente a Deus, por me conceder força e coragem para seguir essa jornada.

A todos os servidores da UTFPR campus Medianeira por proporcionar um ambiente agradável para construção do conhecimento, particularmente secretaria acadêmica e biblioteca que são sempre prestativos e cordiais.

Aos professores que contribuem para formação profissional, transmitindo seus conhecimentos e auxiliando sempre que preciso para dirimir dúvidas e resolver problemas.

Reverencio especialmente aos professores Juliano Lamb, coordenador das atividades de estágio e TCC, sempre solicito nas suas atribuições. A professora Alessandra B. Garbelotti Hoffmann que foi minha orientadora de estágio, por seu carisma e motivação durante o curso. E ao professor Pedro Luiz de Paula Filho meu orientador durante este trabalho de diplomação, grande entusiasta de pesquisas científicas, prático e sincero, transmitindo confiança nas atividades que se engaja e muito me ajudou durante este trabalho dedicando seu tempo para que fosse possível superar as dificuldades.

Agradeço aos colegas de faculdade pelo companheirismo durante os anos de estudo, tornando mais prazerosos estes momentos.

A família e amigos, que sempre estão presentes, pois sem eles o percurso seria moroso e cansativo. E principalmente a minha esposa amada, pelo amor, carinho, dedicação e por nosso filho Miguel.

## RESUMO

BERLANDA, Cleifai A. G. Processamento digital de imagens aplicado à avicultura. 2015 Trabalho de Diplomação (Curso superior de Tecnologia em Análise e desenvolvimento de sistemas). Universidade Tecnológica Federal do Paraná – UTFPR. Medianeira. 2015.

O avanço na área de processamento digital de imagens proporciona ganho de produtividade e respostas consistentes nos mais diversos setores de produção. Tendo este trabalho o objetivo de utilizar técnicas de processamento digital de imagens para contar o número de objetos em cena, foram utilizadas operações aritméticas como subtração, soma e multiplicação, técnicas da morfologia matemática como erosão e dilatação, técnicas de detecção de bordas como *canny*, *findContours* e *drawContours* e a técnica de binarização *inRange*, com o objetivo de contar o número de aves na região de interesse.

Palavras chave: contar objetos, processamento digital de imagens, pré-processamento.

## **RESUMO EM LINGUA ESTRANGEIRA**

BERLANDA, Cleifai A. G. Processamento digital de imagens aplicado à avicultura. 2015 Trabalho de Diplomação (Curso superior de Tecnologia em Análise e desenvolvimento de sistemas). Universidade Tecnológica Federal do Paraná – UTFPR. Medianeira. 2015.

The advancement in image digital processing area provides gains in productivity and consistent responses in various production sectors. With this work the aim of using image digital processing techniques to count the number of objects in the scene, techniques were used arithmetic operations such as subtraction, addition and multiplication techniques of mathematical morphology such as erosion and dilation, edges detection techniques as canny, findContours and drawContours and binarization technique inRange, in objective to count the number of birds in the region of interest.

Keywords: count objects, digital image processing, pre-processing.

## LISTA DE FIGURAS

FIGURA 1 – ETAPAS DE UM SISTEMA DE PROCESSAMENTO DE IMAGENS.....	14
FIGURA 2- EXEMPLO DE UTILIZAÇÃO DO ROI.....	15
FIGURA 3 - EXEMPLO DA UTILIZAÇÃO DO ROI.....	16
FIGURA 4 - EXEMPLO DA OPERAÇÃO EROÇÃO.....	17
FIGURA 5- IMAGEM EM TONS DE CINZA.....	18
FIGURA 6- IMAGEM BINARIZADA COM O <i>INRANGE</i> .....	18
FIGURA 7- IMAGEM ORIGINAL.....	23
FIGURA 8 - IMAGEM UTILIZADA PARA GERAR O MODELO.....	23
FIGURA 9 - RESULTADO DA SUBTRAÇÃO DO MODELO POR UMA IMAGEM COM AVES ADULTAS.....	24
FIGURA 10 - MODELO EDITADO PARA REMOVE OBJETOS QUE GERAM RUÍDOS.....	24
FIGURA 11 - IMAGEM APÓS SUBTRAÇÃO DE FUNDO E <i>INRANGE</i> .....	25
FIGURA 12 - IMAGEM SEM OPERAÇÕES ANTES DO <i>INRANGE</i> .....	26
FIGURA 13 - <i>INRANGE</i> APÓS MULTIPLICAÇÃO POR 0.7.....	26
FIGURA 14 - RESULTADO DO <i>CANNY</i> E <i>FINDCOUNTOURS</i> .....	27
FIGURA 15 - RESULTADO DA SUBTRAÇÃO.....	27
FIGURA 16 - RESULTADO DA SUBTRAÇÃO BINARIZADA E SEU OPOSTO.....	28
FIGURA 17 - IMAGEM APÓS SEQUENCIA DE EROSÕES E DILATAÇÕES.....	28
FIGURA 18 - AVES PRETAS MAIS AVES ERODIDAS BRANCAS E O RESULTADO DA SOMA DELAS.....	29
FIGURA 19 - RESULTADO DA APLICAÇÃO DO <i>CANNY</i> E <i>FINDCOUNTOURS</i> NA IMAGEM ERODIDA.....	29
FIGURA 20 - CONSOLE APRESENTANDO A QUANTIDADE DE AVES.....	31
FIGURA 21 - JANELA QUE POSSIBILITA O USUÁRIO INTERAGIR COM O SISTEMA.....	31
FIGURA 22 - TODAS AS JANELAS DO SISTEMA.....	32
FIGURA 23 - IMAGEM DAS TELAS DO SISTEMA PARA VALIDAÇÃO.....	34
FIGURA 24 - DEFINIÇÃO DA ÁREA DA AVE DE ACORDO COM A POSIÇÃO DA MESMA NA IMAGEM.....	35
FIGURA 25 - VALIDAÇÃO DO RESULTADO.....	36
FIGURA 26 - MÉDIA E MEDIANA DE 2880 FRAMES.....	37
FIGURA 27 - CÁLCULO DA MÉDIA E MEDIANA COM 7200 FRAMES.....	38
FIGURA 28 - CÁLCULO DA MÉDIA E MEDIANA COM 14400 FRAMES.....	40
FIGURA 29 – 2880 FRAMES.....	41
FIGURA 30 – IMAGEM DO ARQUIVO COM TODAS AS COLUNAS.....	48
FIGURA 31 – IMAGEM DE PARTE DO ARQUIVO GERADO.....	48



## LISTA DE SIGLAS

BSD	<i>Berkely Software Distribution</i>
COADS	Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas
GCC	<i>GNU Compiler Collection</i>
Embrapa	Empresa Brasileira de Pesquisa Agropecuária
GNU	GNU's Not Unix
IDE	<i>Integrated Development Environment</i>
FPS	<i>Frames por Segundos</i>
MinGW	Minimalist GNU for Windows
NA	Não Avaliado
<i>OpenCV</i>	<i>Open Source Computer Vision</i>
UTFPR	Universidade Tecnológica Federal do Paraná

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO.....</b>	<b>10</b>
1.1	OBJETIVOS .....	11
1.2	JUSTIFICATIVA .....	11
1.3	ESTRUTURA DO TRABALHO .....	12
<b>2</b>	<b>REFERENCIAL TEÓRICO .....</b>	<b>12</b>
<b>3</b>	<b>ESTADO DA ARTE.....</b>	<b>13</b>
<b>4</b>	<b>METODOLOGIA.....</b>	<b>22</b>
4.1	FORMA DE UTILIZAÇÃO DO SISTEMA.....	30
<b>5</b>	<b>RESULTADOS E DISCUSSÕES.....</b>	<b>33</b>
5.1	SUBTRAÇÃO DE FUNDO, <i>INRANGE</i> E CONTAGEM DE <i>PIXELS</i> .....	33
5.2	<i>CANNY, FINDCOUNTOURS, DRAWCOUNTOURS</i> .....	33
5.3	EROSÃO DILATAÇÃO <i>CANNY, FINDCOUNTOURS, DRAWCOUNTOURS</i> .....	33
5.4	TESTES .....	34
<b>6</b>	<b>CONSIDERAÇÕES FINAIS.....</b>	<b>45</b>
6.1	CONCLUSÃO.....	45
6.2	TRABALHOS FUTUROS .....	45
	<b>REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>46</b>
	<b>ANEXO A – MODELO DO ARQUIVO GERADO PELO SISTEMA.....</b>	<b>48</b>

## 1 INTRODUÇÃO

O desenvolvimento de sistemas autônomos com recursos similares ao sistema visual humano e que seja capaz de reagir a estímulos visuais de forma apropriada à área específica sob investigação, ainda é um grande desafio (PEDRINI e SCHWARTZ, 2008). Entretanto a necessidade desse tipo de sistema é cada vez maior, uma vez que os dispositivos para captura de imagens estão mais sofisticados e mais acessíveis, tornando possível que funções desempenhadas de forma manual possam ser realizadas de forma automatizadas.

Nesse sentido a visão computacional busca auxiliar na resolução de problemas, imitando a cognição humana e a habilidade de tomar decisões de acordo com as informações contidas nas imagens. A visão computacional é dividida em dois níveis de abstração, processamento de imagens e análise de imagens (PEDRINI e SCHWARTZ, 2008).

Conforme (GOMES,2001) processamento de imagem consiste na utilização de operações matemáticas para alterar valores dos *pixels* (menor área de iluminação em uma tela) de imagens digitais, modificando-as, para facilitar sua visualização e/ou para proceder à extração de dados quantitativos. (Albuquerque e Albuquerque, 2014) reforçam que o processamento de imagens é uma técnica extremamente dependente do problema que se deseja resolver. Por isso geralmente são desenvolvidas soluções específicas, para o problema levantado.

A análise de imagens é realizada com base nas características dos objetos presentes nas imagens (PEDRINI e SCHWARTZ, 2008), ocorrendo após o processamento dela, possibilitando uma melhor visualização dos dados da imagem. Para (Piteri e Rodrigues, 2011) a análise de imagens é o resultado da conversão de uma imagem em algo de nível mais alto que puramente numérico.

Neste sentido este trabalho tem por objetivo utilizar técnicas de processamento e análise de imagens para contar a quantidade de aves em determinados pontos da imagem. As imagens foram cedidas pela Embrapa (Empresa Brasileira de Pesquisa Agropecuária), mais especificamente a unidade de Concórdia/SC e foram concebidas para avaliar o comportamento das aves de corte durante seu estágio no aviário, atendendo ao anseio da cadeia produtiva de aves, de avaliar e definir os padrões comportamentais delas, para posteriormente propor modelos de auxílio a tomada de decisão.

## 1.1 OBJETIVOS

### Objetivo geral

Desenvolver um sistema que possibilite contar as aves em um vídeo para identificar informações comportamentais.

### Objetivos específicos

- Apresentar e conceituar as ferramentas e técnicas utilizadas;
- Estudos do processamento de imagens, voltado às técnicas de contagens de objetos em cena;
- Desenvolvimento do sistema.

## 1.2 JUSTIFICATIVA

Para um observador tirar conclusões a partir da visualização de uma grande quantidade de imagens provenientes de vídeos, sem o auxílio de programas apropriados, seria maçante, demorado e pouco produtivo para esse profissional, além de comprometer a pesquisa com a percepção do observador, que está suscetível ao cansaço e a interferências do local de trabalho.

(MOGAMI, 2009) menciona que um sistema limita os erros ocasionados pela subjetividade humana pode trabalhar continuamente, e permite alta velocidade no processamento das informações oferecendo respostas mais ágeis.

Contrastando com os aspectos da observação humana, o sistema extrai as informações desejadas com precisão mensurável, consumindo pouco tempo da pessoa responsável por operá-lo e libera o profissional que estaria levantando as informações *in loco* para que ele possa analisar as informações extraídas com o sistema. E possibilita refazer o estudo, desde que utilizando vídeos com as características iguais aos que foram utilizados para o desenvolvimento do sistema.

Tendo em vista as discrepâncias entre as duas opções apresentadas para extração de dados dos vídeos em questão, considera-se viável o desenvolvimento de um sistema que possa efetuar a contagem das aves na cena, em partes definidas pelo operador do sistema através do mouse.

### 1.3 ESTRUTURA DO TRABALHO

O presente trabalho está dividido em seis capítulos. No primeiro é explanado sobre a área de atuação deste projeto os objetivos dele e a justificativa para sua implementação. O segundo capítulo apresenta o referencial teórico que serve para situar o leitor sobre os recursos utilizados. O terceiro trata do estado da arte, descrito para apresentar outros trabalhos relacionados cujo conteúdo agregou conhecimento para o desenvolvimento desse projeto. No quarto capítulo é apresentada a metodologia adotada para execução do projeto, onde é descrito os materiais e métodos utilizados. O quinto apresenta os resultados e discussões para cada um dos métodos utilizados e os casos de testes para o método que possibilitou aproveitamento do sistema. O sexto capítulo apresenta a conclusão do trabalho e as expectativas de trabalhos futuros relacionados a essa área de conhecimento. Após há as referencias bibliográficas utilizadas no desenvolvimento desse projeto. E por ultimo no anexo “A”, duas imagens referentes ao arquivo gerado pelo sistema com os resultados do processamento de um vídeo.

## 2 REFERENCIAL TEÓRICO

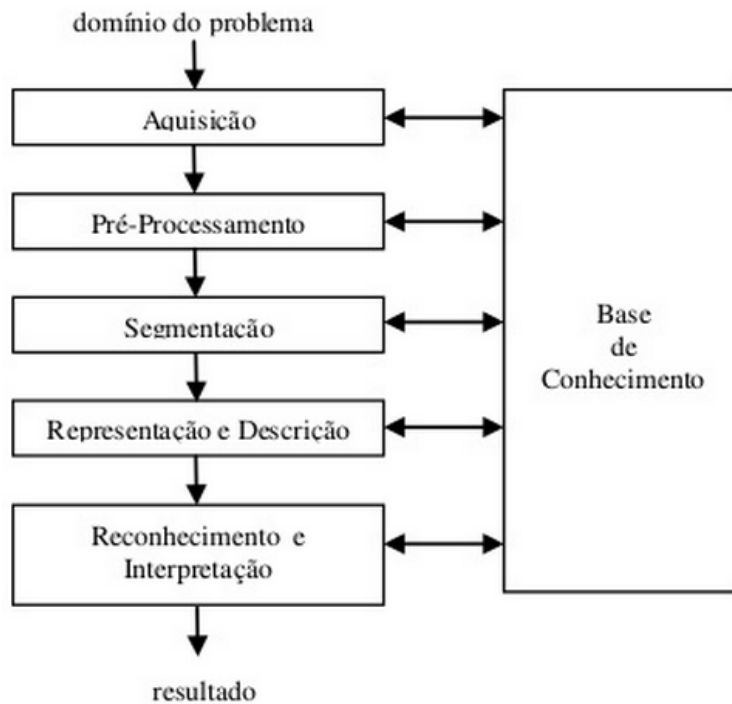
Uma imagem é uma representação gráfica, plástica ou fotográfica de um objeto (FACON 1996), ou ainda, segundo (GONZALEZ e WOODS, 2010) é uma função bidimensional,  $f(x,y)$ , em que  $x$  e  $y$  são coordenadas espaciais e a amplitude de  $f$  em qualquer par de coordenadas é chamada de intensidade ou nível de cinza da imagem. Essas duas definições são complementares, pois a primeira diz respeito à visão humana, enquanto a segunda é a forma com a qual é possível manipular uma imagem digital.

Existem diversas formas de se manipular imagens digitais, cada uma apropriada para cada situação onde é necessário o processamento digital de imagens. Nesse estudo será abordado à manipulação de imagens digitais, para reduzir ruídos e extrair dados das amostras de imagens disponibilizados para o desenvolvimento desse projeto.

As imagens trabalhadas são partes de vídeos, pois um vídeo nada mais é que uma sequência de imagens denominadas *frames* ou quadros, reproduzidos a uma velocidade conhecida por FPS (*frames* por segundos) (VEIT e ARAUJO, 2014). Exibindo várias imagens consecutivas em um curto espaço de tempo, cria-se a sensação de movimento. E quanto mais imagens no mesmo espaço de tempo, mais natural será essa sensação (MACHADO, 2014).

Segundo (PEDRINI e SCHWARTZ, 2008) processamento digital de imagens é um conjunto de técnicas para capturar, representar e transformar imagens digitais com o auxílio do computador. Enquanto extrair dados de uma imagem está relacionado, a análise de dados. (GONZALEZ e WOODS, 2010) descrevem que análise digital de imagem está situada entre o processamento digital de imagens e a visão computacional, tendo o objetivo de interpretar as informações extraídas da imagem, ou mesmo simplesmente interpretar as informações que estão em uma imagem. (GONZALEZ e WOODS, 2010) dizem ainda, que a visão computacional tem como meta, utilizar computadores para emular a visão humana, inclusive com a capacidade de aprendizado e de agir com base em informações visuais.

(PEDRINI e SCHWARTZ, 2008) dividem o processamento digital de imagens em seis etapas, conforme Figura 1.



**Figura 1 – Etapas de um sistema de processamento de imagens.**

**Fonte: (PEDRINI e SCHWARTZ, 2008).**

Segundo (MARQUES FILHO e VIEIRA NETO, 1999), domínio do problema é questão a ser respondida após o processamento. O primeiro passo na metodologia é a aquisição da imagem. Essa por sua vez pode ser adquirida por qualquer máquina fotográfica e depois passar por um processo de conversão para o formato digital.

O pré-processamento visa melhorar a qualidade da imagem por meio de técnicas próprias para tal. Melhorando a visualização dessa imagem ou facilitando a extração de informações das imagens objetos de estudo (PEDRINI e SCHWARTZ, 2008).

A segmentação consiste em separar o que é importante na imagem do restante dela. A ideia é fazer com que objetos ou áreas de interesse tenham os seus *pixels* com as mesmas propriedades, agrupados e destacados dos demais (PITERI e RODRIGUES, 2011).

Na representação e descrição, são extraídos os dados resultantes da segmentação, podendo apresentar imagens, números ou dados estatísticos. (MARQUES FILHO e VIEIRA NETO, 1999) dizem que os descritores devem ser representados por uma estrutura de dados adequada ao algoritmo de reconhecimento, e que a saída será um conjunto de dados correspondente ao resultado da segmentação.

No reconhecimento e interpretação, o sistema avalia o que os descritores retornaram e executa uma ação com base nessa informação (MARQUES FILHO e VIEIRA NETO, 1999).

Base de conhecimento é referente ao conhecimento que se tem sobre o problema a ser resolvido. (MARQUES FILHO e VIEIRA NETO, 1999).

As técnicas abordadas nesse estudo dizem respeito às etapas de pré-processamento, segmentação e a etapa de representação e descrição.

Para otimizar esse processamento, é possível efetua-lo apenas em uma região, utilizando a técnica conhecida por ROI (*Region of Interest*), descrito por (PAULA FILHO, 2014) como um recurso que permite limitar o processamento de uma imagem a uma região de interesse.

Na Figura 2 é possível ver uma imagem dividida em onze partes, uma delas está sendo ignorada enquanto as outras dez estão delimitadas por linhas. As linhas não fazem parte do ROI, estão ali apenas para que o usuário identifique cada uma das regiões.



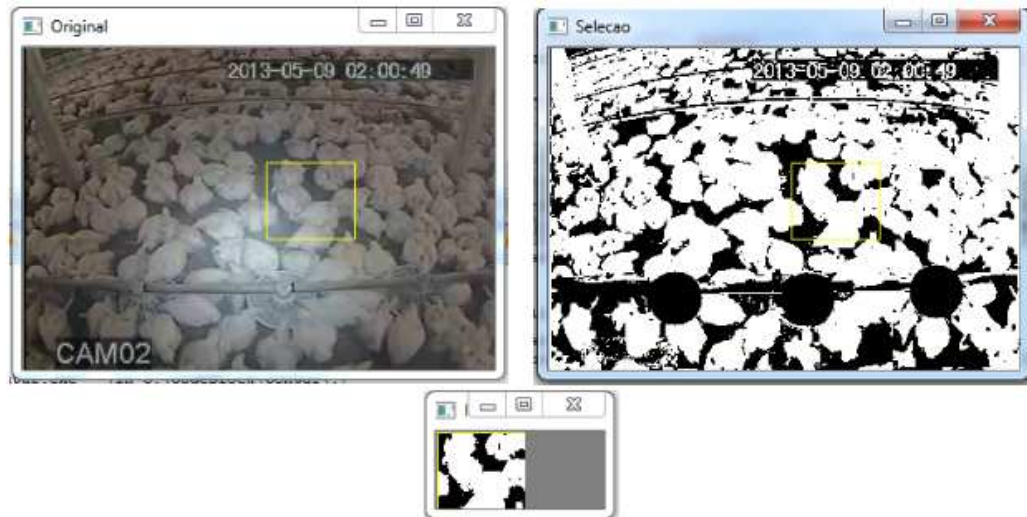
**Figura 2- Exemplo de utilização do ROI.**

**Fonte: Autoria própria**

No exemplo da Figura 2, cada um dos retângulos é uma região de interesse, tratadas de forma individual, cada um com um processamento específico.

A Figura 3, também é outro exemplo da aplicação do ROI, onde foi aplicado um processamento a apenas uma região e foi gerada uma imagem apenas da região de interesse.





**Figura 3 - Exemplo da utilização do ROI.**

**Fonte: Autoria própria.**

Duas das técnicas utilizadas nesse estudo foram erosão e dilatação, mas para entender a ação destas é preciso antes conhecer a morfologia matemática e o elemento estruturante.

Morfologia matemática é o estudo de estruturas matemáticas das entidades presentes em uma imagem, tendo como princípio básico extrair informações relativas à geometria e à topologia de um conjunto desconhecido de uma imagem (FACON, 1996).

A linguagem da morfologia matemática usa a teoria dos conjuntos. Os conjuntos representam os objetos encontrados em uma imagem (GONZALEZ e WOODS, 2011). Esses objetos são chamados de elementos estruturantes. Eles são pequenos conjuntos ou subimagens usadas para examinar uma imagem buscando propriedades de interesse (GONZALEZ e WOODS, 2011). (FACON, 1996) diz que elemento estruturante é um conjunto completamente definido e conhecido (forma, tamanho), que é comparado a partir de uma transformação ao conjunto desconhecido da imagem.

Tendo em mente os conceitos da morfologia matemática e do elemento estruturante é possível compreender o conceito da operação erosão. É uma operação básica, onde a região mais escura cresce sobre a mais clara com o objetivo de eliminar dados de uma imagem que não são semelhantes a um tipo padrão, eliminando os picos positivos mais finos do que o elemento estruturante e expande os picos negativos (PAULA FILHO, 2014).

Na Figura 4 o exemplo da erosão permite ver que a primeira imagem submetida ao elemento estruturante de 3x3 resultou na segunda imagem, afinada. Isso se deve a operação de

erosão que avalia cada *pixel* da imagem, e quando o mesmo não é igual ao *pixel* central do elemento estruturante ele é descartado, ou como no exemplo da Figura 4 convertido para zero.

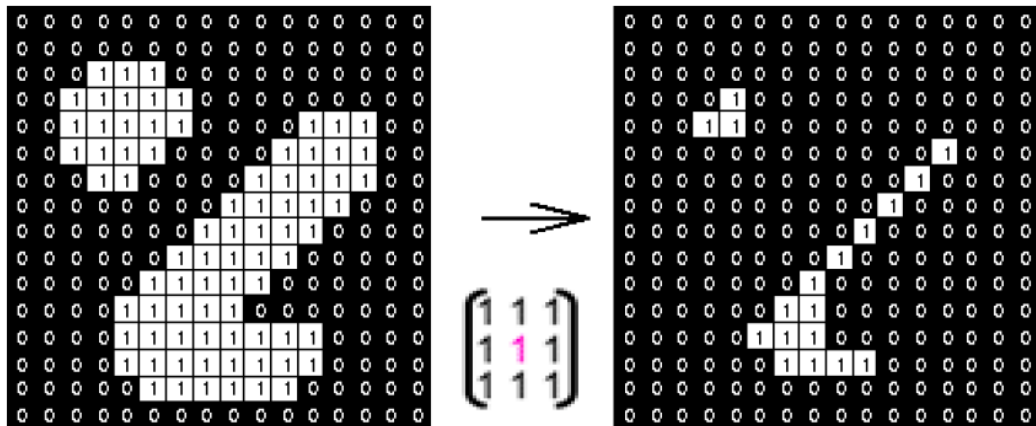


Figura 4 - Exemplo da operação Erosão.

Fonte: (PAULA FILHO, 2014)

A dilatação é a operação contrária a erosão, onde a região mais clara cresce sobre a região mais escura também com base no elemento estruturante (PAULA FILHO, 2014).

Em alguns casos a diversidade de cores e intensidade de brilho na imagem dificulta a extração das informações da imagem. Então para tornar mais fácil o processamento e a visualização dessas informações, a imagem é convertida para tons de cinza, para isso pode ser utilizada a função *cvtColor* que converte uma imagem de um espaço de cor para outro. Após a conversão a imagem pode ser binarizada onde a imagem ficará com apenas duas cores, preto e branco. Para binarizar uma imagem pode-se utilizar a função *thresh* binível *inRange*, função essa que recebendo como parâmetros um vetor, um valor limite mínimo, um valor limite máximo e como quarto parâmetro, outro vetor de destino da operação, este ultimo por sua vez receberá uma imagem em 8 bits, que esta no intervalo informado do primeiro vetor (PAULA FILHO, 2014).

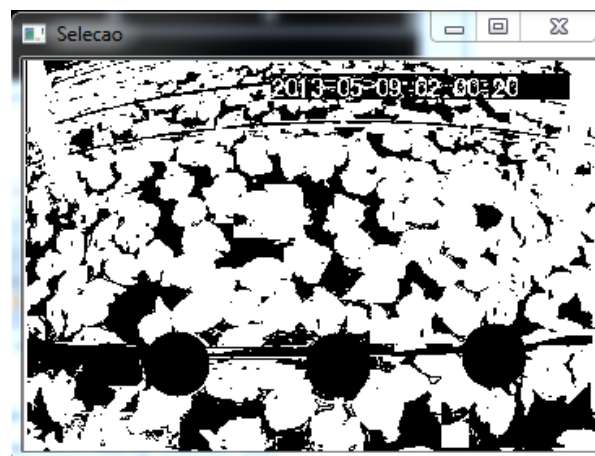
A Figura 5 é a imagem subtraída e convertida para tons de cinza.



**Figura 5- Imagem em tons de cinza.**

**Fonte: Autoria Própria.**

Enquanto a Figura 6 é uma imagem binarizada através do *InRange* (*Threshold* binível) após a subtração do fundo e convertida para tons de cinza.



**Figura 6- Imagem binarizada com o *InRange*.**

**Fonte: Autoria Própria.**

Uma boa forma de distinguir objetos é separá-los através das bordas. Uma borda é o limite entre duas regiões com propriedades distintas de níveis de cinza (PEDRINI e SCHWARTZ, 2008). Segundo (PAULA FILHO, 2014) um dos melhores detectores de bordas é o algoritmo de *Canny*, fazendo uso do filtro Gaussiano, gradiente e *thresholds* para detectar as bordas dos objetos. Entretanto quando os objetos estão encostados uns nos outros e eles possuem coloração semelhante ou quando os mesmos são disformes, não apresentando um padrão, continua sendo um desafio separá-los, pois as bordas dos mesmos não aparecem por ter diversos objetos contínuos dando a impressão de ser um único objeto.

A Figura 5 é resultado da subtração de imagens. A subtração de imagens é uma operação aritmética realizada entre imagens. As operações aritméticas são realizadas *pixel a pixel*, podendo ser realizada entre imagens desde que as imagens envolvidas na operação tenham tamanhos iguais, pois a operação será realizada entre os pares de *pixels* correspondentes (GONZALEZ, WOODS 2010). Também é possível realizar uma das operações entre uma imagem e um número, onde cada *pixel* sofrerá a operação pelo valor informado (PAULA FILHO, 2014).

Média aritmética é o quociente entre a soma dos  $n$  valores dos dados com o número de elementos do conjunto, podendo ser amostral ou populacional, tendo o mesmo procedimento matemático, mas significados diferentes. A média de uma amostra é um parâmetro estimado da verdadeira média da população (MUCELIN, 2003). Enquanto a mediana de um conjunto de dados divide a amostra em duas partes iguais. Para identificar a mediana, é preciso ordenar os valores do menor para o maior, e pegar o valor do meio. A mediana permite identificar a partir de seu valor, que 50% dos elementos do conjunto possuem valores abaixo deste, enquanto que os demais estão acima (MUCELIN, 2003).

(FACON, 1996) fala que o cálculo da área em uma entidade digital, consiste em verificar o número de pontos contidos nessa entidade, podendo esta operação ser interpretada pela quantidade de vezes que um ponto teste, percorrendo a imagem, encontra a entidade estudada.

Segundo (FACON, 1996), *TopHat* consiste em usar uma combinação entre uma imagem original e a imagem correspondente aberta, para a detecção de picos na imagem. (PAULA FILHO, 2014) explica que essa combinação é referente à subtração de uma imagem aberta da sua original, onde imagem aberta ocorre quando a imagem A é erodida pela imagem B e depois o resultado desse processo é erodido pelo mesmo elemento estruturante.

*DrawContours* é uma função para desenhar os contornos das imagens. Essa função pode ser utilizada em conjunto com a função *FindContours*. A função *FindContours* é utilizada para encontrar contornos em imagens binárias. A função *bitwise\_not* aplica operações algébrica binária que inverte todos os bits da matriz informada, resultando em uma imagem contrária ao parâmetro de entrada (PAULA FILHO, 2014).

### 3 ESTADO DA ARTE

Estado da arte apresenta projetos relacionados à pesquisa abordada, possibilitando ao pesquisador conhecimento referente ao nível de desenvolvimento na área em que está atuando além de identificar tecnologias e métodos já empregados e os resultados obtidos. Na sequencia há uma breve descrição de quatro trabalhos relacionados à contagem de objetos. Os mesmos foram selecionados devido ao objeto a ser contado e as condições da filmagem serem similares ao objeto de estudo desse trabalho.

Em (Mogami, 2009) foi desenvolvido um trabalho com o objetivo de avaliar, o bem estar e a massa corporal de frangos de corte através de análise digital de imagens. Para analisar a massa corporal das aves, foi utilizada a binarização da imagem e a contagem dos *pixels*. A equação (1) foi utilizada para determinar a quantidade de aves em volta dos comedouros.

$$N \text{ frangos} = \left( \frac{PT - C}{PF} \right) \quad (1)$$

Onde *N frangos* é igual ao número de frangos na região do comedouro. *PT* corresponde ao número total de *pixels* da região do comedouro. *C* corresponde ao número de *pixels* do comedouro e da tubulação. *PF* corresponde ao número de *pixels* referente a um frango, sendo que *PF* foi definido de forma empírica (MOGAMI, 2009).

Com esta metodologia (MOGAMI, 2009) relatou alcançar resultados satisfatórios no que diz respeito a determinar a massa corporal da ave e a frequência delas nos comedouros e bebedouros.

Já em (SALIS, PEREIRA, 2007) foi realizada a contagem automática de tarugos de aço por meio da visão computacional fazendo uso do operador morfológico *TopHat*, técnica para corrigir a iluminação da imagem, seguido da binarização, com limiar definido de forma empírica e mantendo-se estático para esse trabalho, aliado ao filtro de médias, os autores conseguiram imagens homogêneas, nas quais aplicaram sucessivas operações de erosão e dilatação, a fim de separar os objetos de interesse, chegando a um índice de 96% de acerto.

(PORTELA, 2009) realizou experimentos para contagem de objetos em imagens digitais, mais especificamente ovos de *Aedes Egypti* em imagens de ovitrampas. O método com melhor índice de acerto consiste na binarização da imagem, o fechamento morfológico e a contagem de *pixels*, onde contando o número de *pixels* brancos totais e dividindo pela área média do ovo, estabelecida para esse projeto, tem-se a quantidade de ovos de mosquito na

imagem. Este método se mostrou mais eficaz, do que a contagem manual, entretanto o autor ressalta que podem ser feitas melhorias no mesmo.

(BARBEDO, 2012) apresenta uma revisão de literatura com diversas estratégias, propostas para contagem de objetos, estando elas organizadas cronologicamente dentro de cada assunto, e para cada técnica é apresentado uma explicação detalhada da sua aplicação e uma breve descrição das técnicas utilizadas.

## 4 METODOLOGIA

As imagens utilizadas neste projeto foram coletadas em aviário *dark house* (casa escura) de 100 x 12 m, na região Oeste de Santa Catarina, por oito câmeras infravermelhas que geraram imagens coloridas por 42 dias consecutivos. As câmeras foram distribuídas na lateral do pinteiro a cada 10 m, sendo quatro de cada lado. A altura de instalação das câmeras foi de 1,80 m inclinadas para baixo em ângulo de 45°. À medida que as aves foram crescendo as câmeras foram deslocadas uniformemente nas laterais (ABREU, 2013).

Durante o desenvolvimento do sistema para contagem das aves, foram utilizadas técnicas de detecção de bordas, binarização, erosão, dilatação, soma, subtração, multiplicação, a combinação destas, sendo que os melhores resultados foram obtidos utilizando em conjunto subtração de fundo, *Threshold* e a contagem binária dos *pixels*. Para utilizar a contagem binária dos *pixels*, foi definido com base nesse estudo um valor aproximado referente à área média de uma ave para a imagem, valor esse que varia conforme a posição da ave na imagem.

Para o desenvolvimento do sistema foi utilizado o CodeBlocks, na versão 13.12, GCC 4.7.1, 32 bits, na plataforma Windows 64 bits com os compiladores MinGW e o Cmake na versão 3.0, 32 bits para compilar a biblioteca do OpenCV versão 2.4.9. CodeBlocks é, segundo (CODEBLOCKS, 2014), uma IDE (ambiente de desenvolvimento integrado) de código aberto e multiplataforma, cuja arquitetura é baseada em *plug-ins* (que é modular, acoplável). O que facilita a integração com outros módulos necessários. E para o processamento de imagens foi utilizado a biblioteca OpenCV, *Open Source Computer Vision Library*, disponível sob licença BSD, foi construída para proporcionar infra estrutura para aplicações de visão computacional contendo uma biblioteca de programas de aprendizagem de máquina que somados aos algoritmos de visão computacional, contabilizam mais de 2.500 algoritmos otimizados (ITSEEZ, 2014).

Devido às câmeras serem infravermelhas, a luminosidade gerada por elas não é uniforme, produzindo um clarão no centro das imagens. O clarão é no formato circular e com intensidade variada de forma degrade, ficando as imagens muito claras ao centro e escuras nas laterais. Na Figura 7 é possível ver um exemplo.



**Figura 7- Imagem original.**

**Fonte: (EMBRAPA, 2013).**

Para a visão humana, essa imagem permite identificar as aves com certa tranquilidade, entretanto a variação de brilho, cor e tamanho, para um sistema automatizado torna a atividade mais complexa. Visando resolver o problema da intensidade do brilho de forma desproporcional na imagem, foi utilizada a operação aritmética subtração.

Para poder realizar a subtração de fundo, é necessário ter uma imagem compatível, entretanto, não há uma imagem de fundo para cada um dos vídeos que devem ser submetidos ao processamento, então optou se por utilizar uma imagem de um dos vídeos sem as aves. Porém os vídeos sem as aves foram realizados com as câmeras em posições diferentes das imagens realizadas com as aves, e em alguns dos vídeos havia outros objetos, que não estão nos vídeos que há interesse em avaliar a quantidade de aves, conforme é possível visualizar na Figura 8.



**Figura 8 - Imagem utilizada para gerar o modelo.**

**Fonte: (EMBRAPA, 2013).**



Utilizando a imagem referente à Figura 8 para subtração, gera mais ruídos, devido aos objetos diferentes que tem na imagem modelo, como os pratos e o tonel, e as posições diferentes dos objetos que estão em ambas às imagens, como os comedouros, bebedouros e palanques.



**Figura 9 - Resultado da subtração do modelo por uma imagem com aves adultas.**

**Fonte: Autoria própria.**

Para eliminar esses ruídos, os comedouros e os pratos da imagem modelo, foram substituídos por borrões, que foram gerados do ponto mais próximo ao início dos comedouros pratos e bebedouros, porém antes dos mesmos, para dessa forma manter a cor e o brilho semelhante a do solo das imagens com as aves, enquanto que a parte superior e o tonel tiveram a coloração definida manualmente, pois a seleção do ponto dinâmico, não melhorou a visualização das aves. O resultado pode ser visto na Figura 10.



**Figura 10 - Modelo editado para remover objetos que geram ruídos.**

**Fonte: Autoria própria.**

Uma vez que a subtração de fundo tornou o brilho da imagem mais homogêneo, foi utilizada a função *cvtColor* para converter a imagem para tons de cinza, e assim poder binarizar essa imagem.

A função *inRange* foi testada com diversos parâmetros, e com base nesses testes ficou estabelecido o intervalo entre o menor valor e o maior valor deve ser 50. Também foi constatado que utilizando valores mais baixos o *inRange* difere as aves do solo nas laterais e valores mais altos no centro, isso em imagens sem outros pré-processamento. Foram realizados testes multiplicando a imagem por 0.5, 0.7, utilizando um modelo para reduzir a diferença do brilho na imagem, subtraindo o fundo e sem nenhum pré-processamento, apenas a imagem com as aves, sendo que o melhor resultado foi obtido subtraindo o fundo conforme é possível ver na Figura 11.

A Figura 11(a) é a imagem original, enquanto a Figura 11(b) é o resultado da operação descrita acima.



**Figura 11 - Imagem após subtração de fundo e *inRange*.**  
**Fonte: Autoria própria.**

Utilizando destes métodos a maior parte da imagem ficou uniforme, possibilitando a partir deste resultado, extrair informações sem ter que realizar processamentos por regiões para ter que diferenciar os objetos de interesse do fundo.

Outro teste realizado foi sem o pré-processamento antes de aplicar o *inRange*. Desta forma também houve bons resultados, porém é preciso trabalhar cada região da imagem que possua brilho diferente da área central, conforme é possível visualizar na Figura 12, onde na Figura 12(a) é a imagem original e na Figura 12(b) a imagem processada.



**Figura 12 - Imagem sem operações antes do *InRange*.**  
**Fonte: Autoria própria.**

Multiplicando a imagem por 0.5 e 0.7 para reduzir o brilho dela, antes de aplicar o *inRange* tem resultado similar ao sem pré-processamento, e traz a mesma complexidade de precisar processar cada área com brilho diferente, devido à coloração. O resultado é visualizado na Figura 13. A Figura 13(a) é a imagem original e a Figura 13(b) a imagem processada.



**Figura 13 - *InRange* após multiplicação por 0.7.**  
**Fonte: Autoria própria.**

Não foram realizados testes para identificar os parâmetros do *inRange* necessários para a utilização dos mesmos nas laterais, uma vez que através da subtração de imagens atendeu a toda a imagem.

Realizados os mesmos testes com o algoritmo de *Canny*, entretanto os resultados não obtiveram ganho de desempenho com relação à imagem sem pré-processamento. A utilização deste algoritmo foi prejudicada devido ao ângulo das imagens que não possibilita visualizar onde termina e onde começa uma nova ave, devido às asas, por criar bordas dentro das aves e

pela proximidade uma das outras fazendo com que um grupo de aves pareça um só objeto. Na Figura 14 é possível ver que em alguns pontos o espaço entre as aves, é identificado como sendo um objeto.

A Figura 14 é resultado apenas do uso dos algoritmos *Canny*, *FindContour* e *DrawCountours*, sem nenhum pré processamento antes, na Figura 14(a) é a imagem original, enquanto a Figura 14(b) é o resultado do processamento.



**Figura 14 - Resultado do *Canny* e *FindCountours*.**  
**Fonte: Autoria própria.**

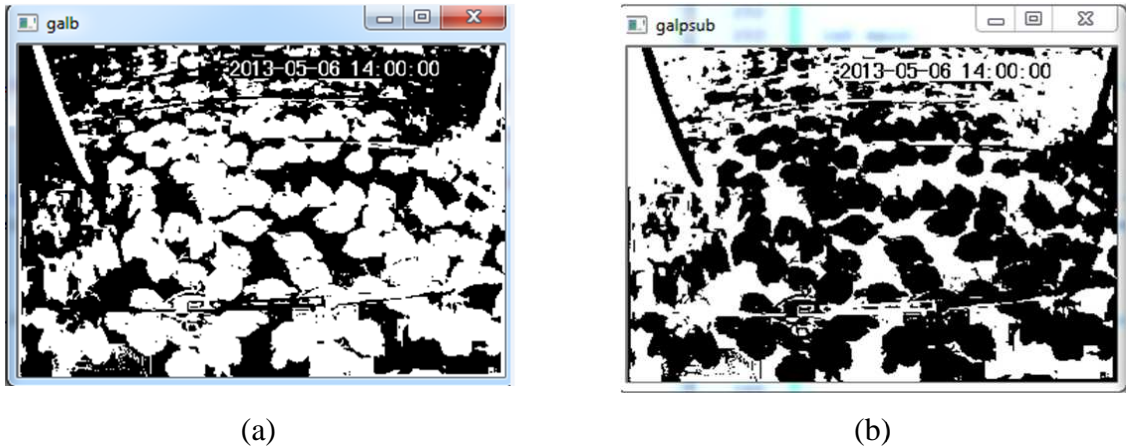
Para tirar proveito da erosão e da dilatação, foi subtraído o fundo da imagem a partir do modelo gerado para esse fim. O resultado desta operação pode ser visto na Figura 15.



**Figura 15 - Resultado da subtração.**  
**Fonte: Autoria própria.**

Após isso a imagem foi binarizada e utilizando a função *bitwise\_not* foi criada uma imagem com a coloração oposta a anterior, tendo assim uma imagem com aves na cor branca

e fundo na cor preta, visualizada na Figura 16(a) e outra imagem com as aves na cor preta e fundo na cor branca, visualizada na Figura 16(b).



**Figura 16 - Resultado da subtração binarizada e seu oposto.**  
**Fonte: Autoria própria.**

Na sequencia a imagem com as aves brancas foi erodida e dilatada, até que os aglomerados tivessem apenas traços, conforme pode ser visto na Figura 17.



**Figura 17 - Imagem após sequencia de erosões e dilatações.**

**Fonte: Autoria própria.**

A imagem que originou a Figura 17 é então somada à imagem com aves pretas Figura 16(b), dessa forma as áreas que tinham aglomeração das aves, terão fendas dentro dos blocos, para possibilitar o desmembramento deles, sem que as aves que estão isoladas sumam completamente. Essa operação pode ser visualizada na Figura 18, os círculos em vermelhos foram feitos para mostrar algumas das fendas provocadas onde havia maior aglomeração das aves.

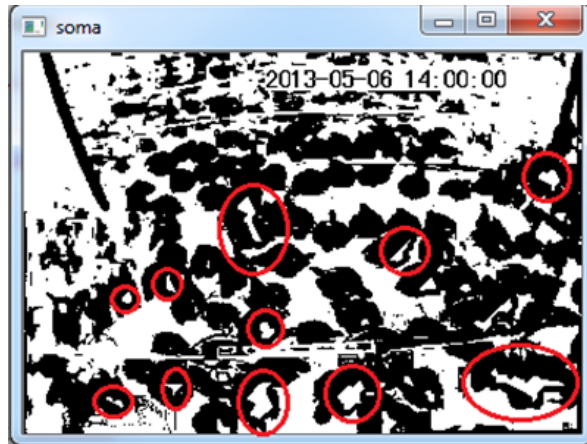
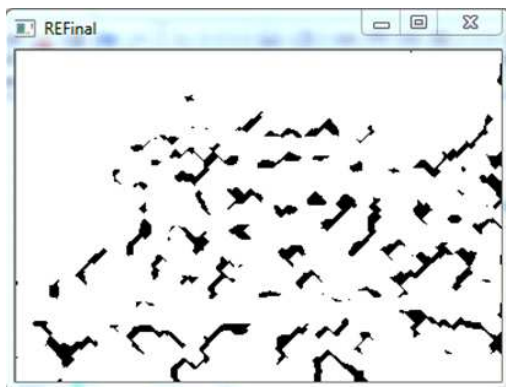


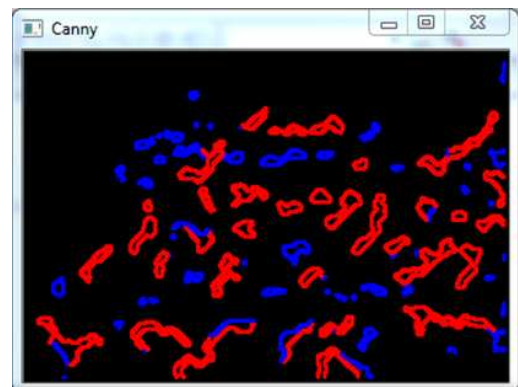
Figura 18 - Aves pretas Mais aves Erodidas brancas e o resultado da soma delas.

Fonte: Autoria própria.

Na sequencia o resultado dessa operação é dilatado para que o solo agora branco cresça sobre as aves separando-as, e erodindo novamente, para que as aves não sumam totalmente. O resultado é visualizado na Figura 19(a).



(a)



(b)

Figura 19 - Resultado da aplicação do *Canny* e *findContours* na imagem erodida.  
Fonte: Autoria própria.

Neste ponto a sequencia de erosões e dilatações foi interrompida e utilizado *Canny* e *findContours* para identificar as bordas, e através do atributo *Hierarchy* saber quantos objetos há na imagem. O resultado pode ser visualizado na Figura 19(b).

Para dirimir a dúvida com relação sobre qual cor seria apropriada para solo e ave, o mesmo processo foi feito com as colorações inversas, entretanto foi obtido o mesmo resultado, pois o que é apresentado são somente as bordas, não diferenciando interior do objeto do exterior, por não serem totalmente identificados.

A partir do resultado obtido através da subtração de imagens e do *inRange*, as aves foram contadas, considerando a quantidade de *pixels* brancos na área selecionada da imagem e dividindo pela área das aves, tendo esta sido definida de modo empírico. Nos locais onde há comedouros foi inserido sobre os mesmos um círculo na cor preta, pois os mesmos tendem a ficar na cor branca comprometendo o cálculo. Em pontos diferentes das imagens as aves apresentam tamanhos variados, o que obriga o sistema aplicar diferente área para as aves onde elas parecem menores, ou devido à subtração e binarização houve perda de parte das aves.

A média e a mediana foram definidas de acordo com o valor da contagem dos *pixels*, a cada 49 *frames*, posterior a quantidade de *frames* o valor de ambas é atualizado. O cálculo da quantidade é realizado a cada *frame*. Para calcular a média o valor do cálculo de cada *frame* é somado em uma variável que foi inicializada com valor zero e ao final de 49 *frames* o valor total dessa variável é dividido por 49, para se ter a média. Já para definir a mediana, o valor do cálculo de cada *frame* é armazenado em um vetor e ao final dos 49 *frames* esse vetor é ordenado em uma sequência crescente e apresentado o valor da posição central do vetor.

Essa quantidade de *frames* pode ser alterada no código fonte do sistema, alterando o valor de três variáveis. Esse procedimento foi executado para verificar qual a quantidade de *frames* possibilita ter um valor de média e mediana mais próxima do número real de aves na cena.

#### 4.1 FORMA DE UTILIZAÇÃO DO SISTEMA

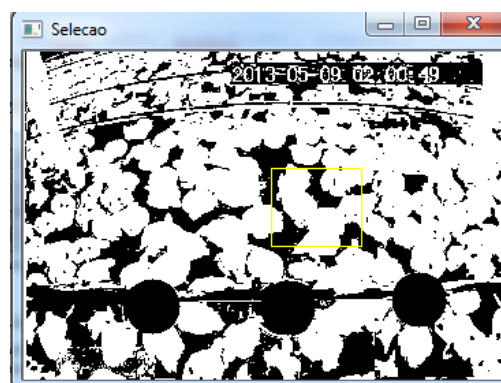
Para utilizar o sistema é preciso ter a imagem de fundo e o vídeo a ser analisado. O sistema apresenta os resultados no console, dividido em duas partes. A lateral esquerda apresenta o resultado da seleção feita pelo usuário, na janela que mostra o vídeo binarizado, e na lateral direita do console, é visto a informação referente à janela com os retângulos delimitando as regiões. Na Figura 20 é possível ver o console durante a execução do sistema.

Resultado da area selecionada			Resultado dos retangulos			
Quantidade	Mediana	Media	R	Quantidade	Mediana	Media
8	8	8	01	7	7	7
			02	9	9	9
			03	11	10	10
			04	11	10	10
			05	13	12	12
			06	27	27	27
			07	13	13	13
			08	6	6	6
			09	18	16	16
			10	5	6	5

**Figura 20 - Console apresentando a quantidade de aves.**

**Fonte: Autoria própria.**

No primeiro clique do botão direito do mouse, o sistema reduz a velocidade que está passando o vídeo e no segundo clique volta à velocidade inicial do mesmo. A janela na qual é possível o usuário interagir com o sistema através do mouse é a janela com a imagem binarizada, com nome “Seleção”. Na Figura 21 essa janela pode ser visualizada.



**Figura 21 - Janela que possibilita o usuário interagir com o sistema.**

**Fonte: Autoria própria.**

Para o usuário selecionar uma área basta dar dois cliques com o botão esquerdo do mouse dentro da janela “Seleção”. O espaço entre o primeiro e o segundo clique será o espaço selecionado para o sistema contar o número de aves. Esse espaço ficará selecionado e marcado em amarelo, na janela que foi selecionada e na janela com a imagem normal, para que o usuário possa validar a informação retornada pelo sistema. O sistema ainda cria uma terceira janela apenas com a área selecionada (ROI). A área da ave, para realizar o cálculo, é definida pelo local do primeiro clique. A quantidade de aves no espaço selecionado será mostrada no console na lateral esquerda. Na Figura 22, é possível visualizar todas as janelas do sistema aberto, após a seleção de uma área com o mouse.



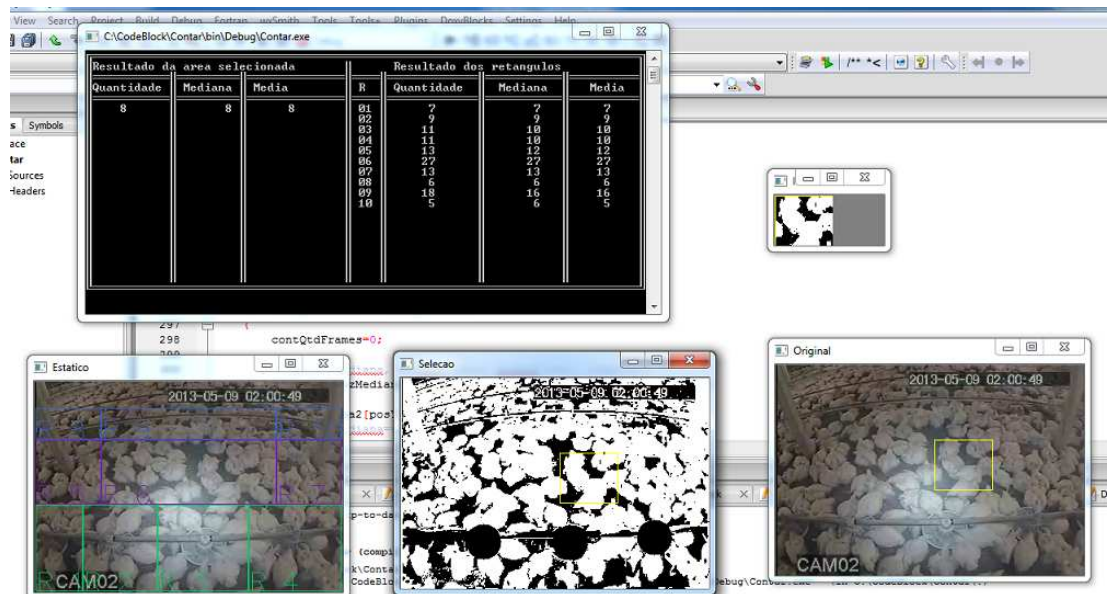


Figura 22 - Todas as janelas do sistema.

Fonte: Autoria própria.

Conforme pode ser visto na Figura 22, há uma janela com nome “Estático”, essa janela tem seu interior demarcado por retângulos, e para cada um deles o sistema tem um valor de área para as aves, o que permite diminuir a margem de erro do cálculo da quantidade de aves. Os retângulos estão identificados pela letra “R” e por um número, e no console na lateral direita existe uma coluna denominada “R” e as linhas com as numerações de cada retângulo. Essa abordagem foi feita para evitar o problema de selecionar uma área da imagem, que é uma intersecção para o valor estabelecido como área da ave. Problema esse que não foi solucionado quando o usuário seleciona uma área do vídeo com o mouse.

O cálculo do número de aves em si, é na verdade uma média dos pixels brancos pela área definida para cada ave. Para reforçar esse valor foi realizada também a média dele e a mediana, levando em consideração 49 frames, que são dois segundos mais um frame. Logo esses valores são atualizados a cada 49 frames. No capítulo resultados e discussões, é possível ver os casos de teste e quantidade de aves contadas para diferentes tamanhos de área, e diferentes quantidades de média e mediana.

Os resultados também são salvos em um arquivo com extensão CSV e cabeçalho para identificação dos campos e posterior análise desses dados. O modelo desse arquivo pode ser visto no anexo A.

## 5 RESULTADOS E DISCUSSÕES

### 5.1 SUBTRAÇÃO DE FUNDO, *INRANGE* E CONTAGEM DE *PIXELS*.

A combinação dessas técnicas possibilitou bons resultados, principalmente quando trabalha com áreas fixas para o cálculo da quantidade de aves. Quando o usuário seleciona a área para o cálculo, o sistema pode apresentar uma margem de erro maior, uma vez que a área da ave utilizada será referente ao local do primeiro clique com o mouse.

Um problema de calcular a quantidade de objetos utilizando a contagem de *pixels*, é que ela na verdade é uma média dos *pixels*, e não distingue objetos. Isso significa que duas metades de uma ave contam como uma ave inteira, então em uma região selecionada que contenha duas aves parcialmente e uma terceira ave inteira, o sistema pode contar como uma ave, duas ou três dependendo de quanto cada uma delas está na área do cálculo. Aumentando a área de interesse, este valor tende a ser mais distante do real, pois haverá mais aves selecionadas parcialmente.

### 5.2 *CANNY*, *FINDCOUNTOURS*, *DRAWCOUNTOURS*.

Apesar de essas técnicas cumprirem muito bem seu objetivo de determinar as bordas dos objetos nas imagens, elas foram ineficazes para contar objetos nos testes realizados, uma vez que existe muita aglomeração das aves, e o ângulo de visão é desfavorável fazendo com que várias aves juntas apareçam como apenas uma. Outro ponto que prejudicou a utilização dessas técnicas foi as asas das aves, pois criam bordas dentro da própria ave, impedindo o sistema de identificar como um objeto único, devido à borda interna.

Logo a utilização dessa técnica para esse projeto foi descartada, por apresentar resultados muito distantes dos reais.

### 5.3 *EROSÃO*, *DILATAÇÃO*, *CANNY*, *FINDCOUNTOURS*, *DRAWCOUNTOURS*.

Diante do resultado para este projeto, de aplicar unicamente as técnicas de detecção de bordas, foram introduzidas à erosão e a dilatação antes da detecção de bordas, para dessa forma, separar os conglomerados de aves, entretanto não houve êxito, pois ao ser agressivo com essas técnicas para separá-las, acaba por eliminar as aves que estão sozinhas. Outra dificuldade encontrada foi que o parâmetro *hierarchy* do *FindCountours*, identificou as bordas dos objetos restantes, como vários contornos distintos, inclusive quando era apenas um objeto, resultando em valores muito acima do real. Essa técnica também se mostrou inviável

para trabalhar por regiões, uma vez que as aves se deslocam e em determinado momento pode haver várias aves numa determinada região, e em outro momento no mesmo local pode estar apenas uma ave, que sendo erodida com intensidade igual a do momento que havia várias aves, faria com que ela sumisse totalmente.

#### 5.4 TESTES

Para verificar a confiabilidade do sistema foi gerada uma imagem do monitor contendo o programa rodando com as telas abertas, conforme Figura 23. E na sequência foi realizada a contagem manual de cada uma das áreas e conferido com o resultado do programa. Lembrando que os valores da média e mediana só mudam depois de decorridos a quantidade de *frames* definida para o teste. Aqui é importante observar que no teste a contagem manual é feita em cima de um único *frame*, enquanto a média e a mediana leva em consideração *N frames*. O vídeo analisado é identificado pelo nome 116\_02\_R\_130509020000.avi cedido por (EMBRAPA, 2013).

As regiões 8 e 10 não possibilitaram a contagem a olho nu, e por isso nas tabelas de resultados consta a sigla NA de não avaliado.

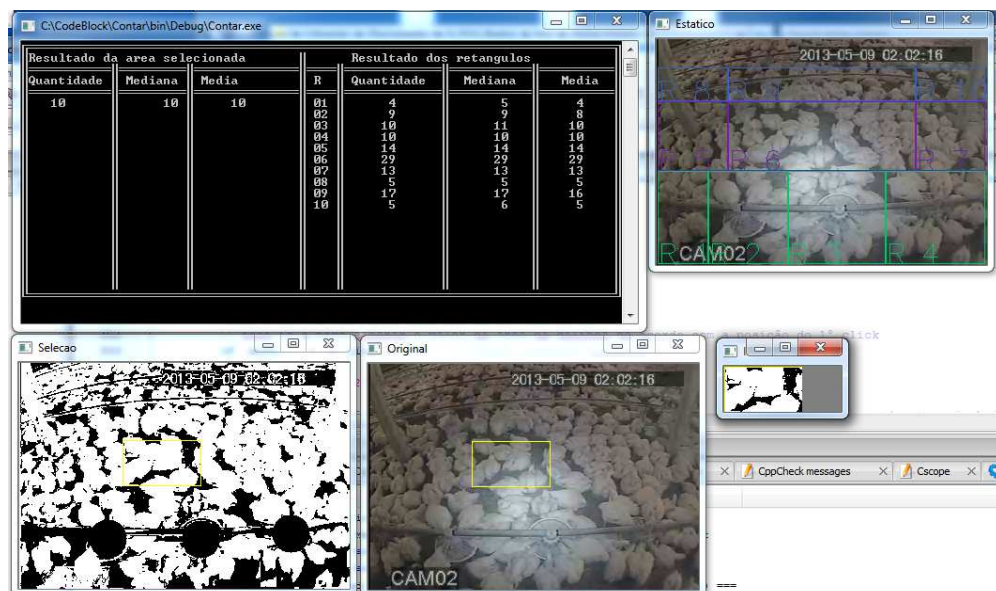


Figura 23 - Imagem das telas do sistema para validação.

Fonte: Autoria própria.

A quantidade de *pixels* utilizada como área de referência de uma ave para calcular a quantidade de aves em um retângulo pode ser visto na Tabela (1).

**Tabela 1 – Área em pixel para uma ave em cada um dos retângulos.**

Retângulo	Área em pixel
1	650
2	477
3	513
4	521
5	281
6	391
7	341
8	291
9	291
10	301

**Fonte: Autoria própria.**

Os parâmetros para área selecionada através do mouse é visível na Figura 24. Foram definidos seis valores. O sexto valor é para os casos do primeiro clique do mouse não esteja dentro de uma das posições definidas anteriormente.

```

352  * para ajustar o valor da area da caixa de seleção com a posição do 1º click
353  ! (xini < 75 && yini >= 140 ) // canto inferior esquerdo
354  area = 361; // 650
355  .se if ( xini >= 270 && yini >= 140 ) // canto inferior direito.
356  area = 453;
357  .se if (xini >= 75 && yini < 270 && yini >= 140) //min entre os dois cantos inferiores (parte mais baixa, comedores).
358  area = 523; //
359  .se if (xini < 90 && yini < 140 && yini >= 90 ) // lateral esquerda, do palanque até a area definida para o canto inferior esquerdo.
360  area = 323;
361  .se if (xini > 90 && yini < 140 && yini >= 90 ) // area central. centro e lateral direita.
362  area = 493;
363  .se if (xini >= 90 && yini < 90 && yini >= 45 && xini < 310) // area superior entre os palanques.
364  area = 277;
365  .se
366  area = 291;

```

**Figura 24 - Definição da área da ave de acordo com a posição dela na imagem.**

**Fonte: Autoria própria.**

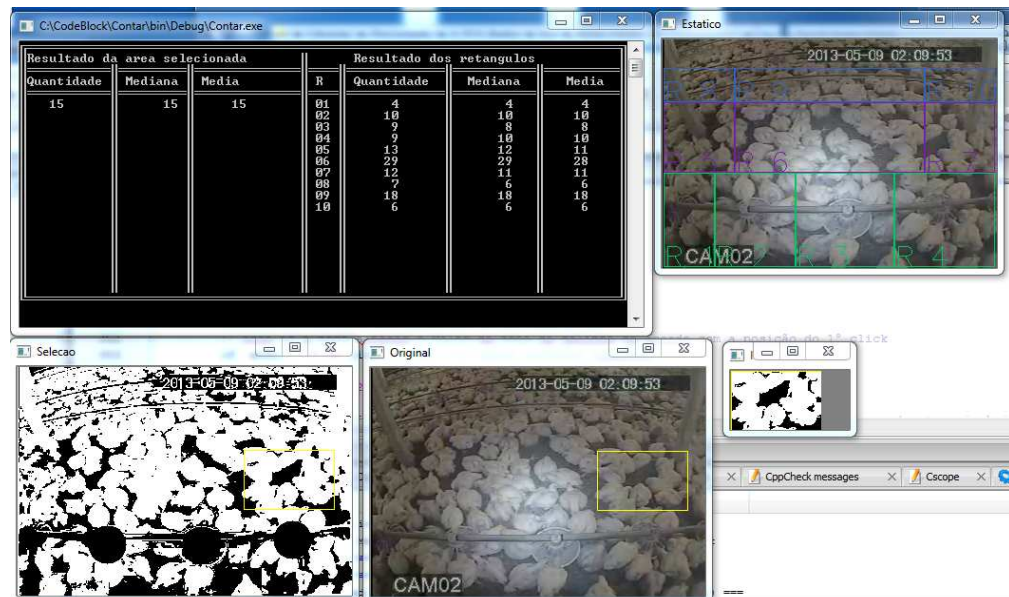
A Figura 23 foi analisada e resultado da contagem pode ser visto na Tabela (2).

**Tabela 2 – Comparativo entre a contagem automatizada e a manual referente à Figura 23.**

Área	Contagem Humana	Contagem algoritmo	Margem de erro %	Mediana	Margem de erro %	Média	Margem de erro %
Manual	8	10	25	10	25	10	25
1	7	4	42,85	5	28,57	4	42,85
2	8	9	12,5	9	12,5	8	0
3	11	10	9,09	11	0	10	9,09
4	13	10	23,07	10	23,07	10	23,07
5	14	14	0	14	0	14	0
6	29	29	0	29	0	29	0
7	11	13	18,18	13	18,18	13	18,18
8	NA	5	NA	5	NA	5	NA
9	16	17	6,25	17	6,25	16	0
10	NA	5	NA	6	NA	5	NA
Erro Médio			15,21		12,61		13,13

**Fonte: Autoria própria.**

Utilizando o mesmo vídeo, porém por um tempo de avaliação maior será realizado o mesmo comparativo na Figura 25.



**Figura 25 - Validação do resultado.**

**Fonte: Autoria própria.**

O resultado do comparativo da Figura 25 pode ser visto na Tabela (3).

**Tabela 3 – Comparativo entre a contagem automatizada e a manual referente à Figura 25.**

Área	Contagem Humana	Contagem algoritmo	Margem de erro %	Mediana	Margem de erro %	Média	Margem de erro %
Manual	13	15	15,38	15	15,38	15	15,38
1	5	4	20	4	20	4	20
2	10	10	0	10	0	10	0
3	8	9	12,5	8	0	8	0
4	9	10	11,11	10	11,11	9	0
5	10	13	30	12	20	11	10
6	29	29	0	29	0	28	3,44
7	9	12	33,33	11	22,22	11	22,22
8	NA	6	NA	6	NA	7	NA
9	23	18	21,73	18	21,73	18	21,73
10	NA	6	NA	6	NA	6	NA
Erro médio			16,00		12,27		10,30

**Fonte: Autoria própria.**

A Figura 26 teve a quantidade de *frames* para cálculo da média e mediana alterados para 2880 *frames* equivalentes a dois minutos do vídeo.



**Figura 26 - Média e mediana de 2880 frames.**

**Fonte: Autoria Própria**

O resultado do teste com a contagem manual pode ser visto na Tabela 4

**Tabela 4 – Comparativo entre a contagem automatizada e a manual referente à Figura 26.**

Área	Contagem Humana	Contagem algoritmo	Margem de erro %	Mediana	Margem de erro %	Média	Margem de erro %
Manual	8	10	25	8	0	8	0
1	6	4	33,33	4	33,33	3	50
2	8	9	12,5	9	12,5	9	12,5
3	8	10	25	11	37,5	10	25
4	10	11	10	10	0	10	0
5	15	13	13,33	13	13,33	13	13,33
6	30	29	3,33	28	6,67	27	10
7	11	12	9,09	13	18,18	12	9,09
8	NA	6	NA	6	NA	5	NA
9	24	18	-25	17	29,16	17	29,16
10	NA	5	NA	6	NA	5	NA
Erro médio			17,39		16,74		16,56

Fonte: Autoria própria.

A Figura 27 teve a quantidade de *frames* alterada para 7200 *frames*, equivalendo a cinco minutos do tempo do vídeo.

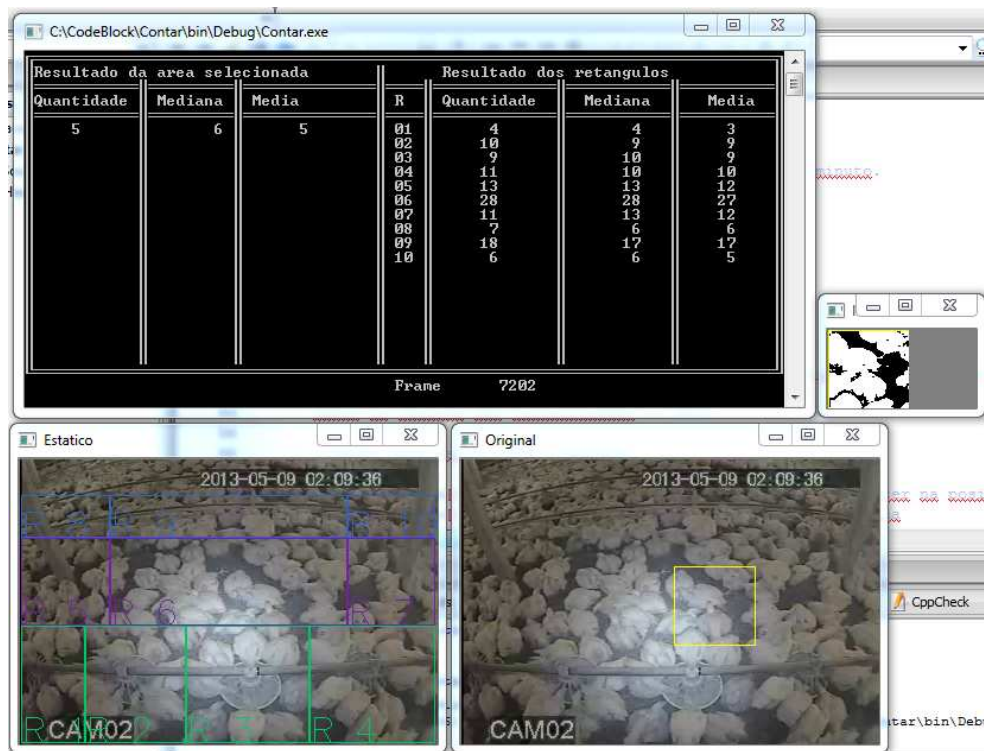


Figura 27 - Cálculo da Média e mediana com 7200 frames.

Fonte: Autoria Própria.

O resultado da análise da Figura 27 pode ser visto na Tabela 5.

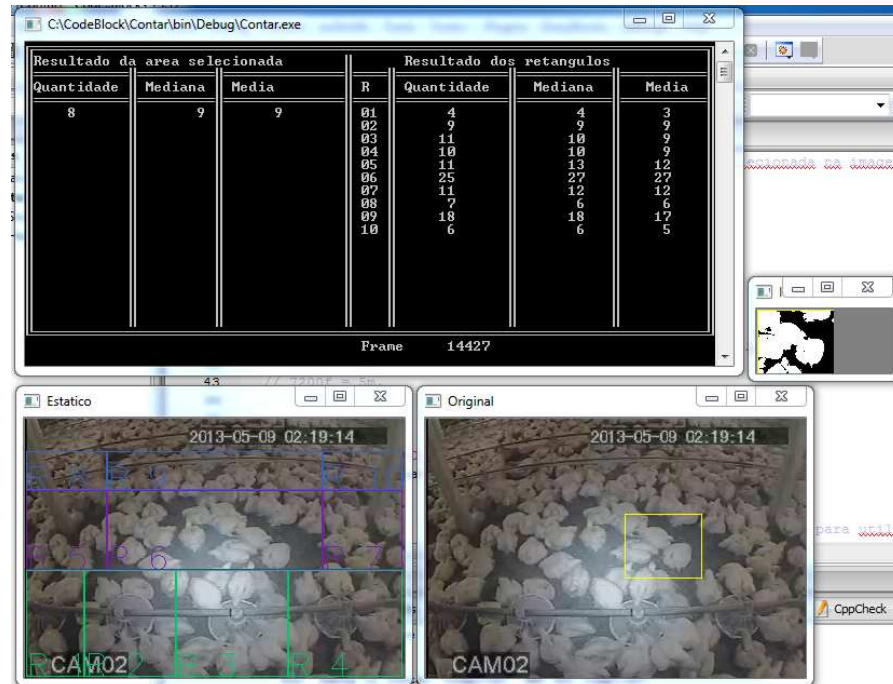
**Tabela 5 – Comparativo entre a contagem automatizada e a manual referente à Figura 27.**

Área	Contagem Humana	Contagem	Margem de erro %	Mediana	Margem de erro %	Média	Margem de erro %
Manual	6	5	16,67	6	0	5	16,67
1	5	4	20	4	20	3	40
2	8	10	25	9	12,5	9	12,5
3	8	9	12,5	10	25	9	12,5
4	10	11	10	10	0	10	0
5	10	13	30	13	30	12	20
6	28	28	0	28	0	27	3,57
7	10	11	10	13	30	12	20
8	NA	7	NA	6	NA	6	NA
9	19	18	5,26	17	10,52	17	10,52
10	NA	6	NA	6	NA	5	NA
Erro médio			14,38		14,22		15,08

**Fonte: Autoria própria.**

A Figura 28 teve a quantidade de *frames* para cálculo da média e mediana alterada para 14400 *frames*.





**Figura 28 - Cálculo da média e mediana com 14400 frames.**

**Fonte: Autoria Própria.**

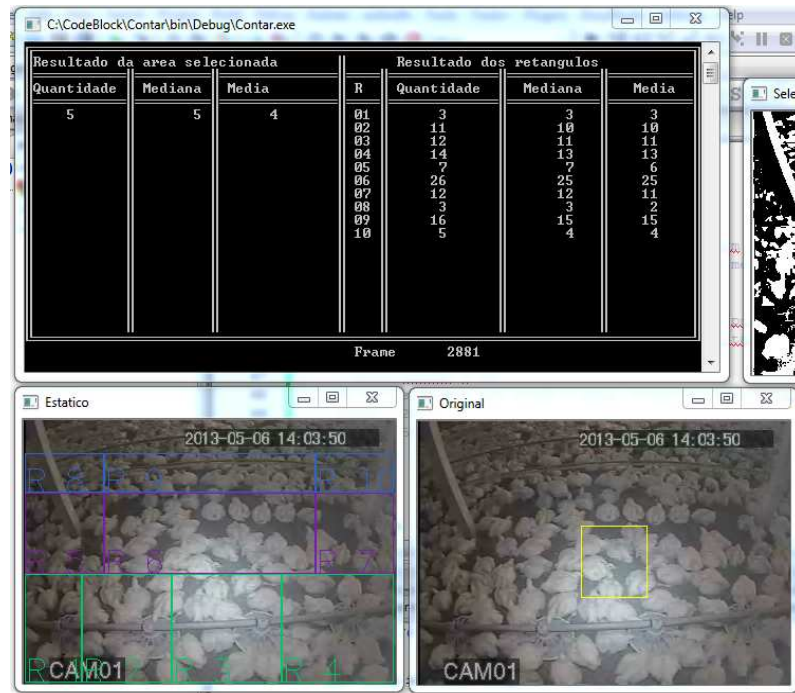
O resultado da análise da Figura 28 pode ser visto na Tabela 6.

**Tabela 6 – Comparativo entre a contagem automatizada e a manual referente a Figura 28.**

Área	Contagem Humana	Contagem Algoritmo	Margem de erro %	Mediana	Margem de erro %	Média	Margem de erro %
Manual	4	8	100	9	125	9	125
1	3	4	33,33	4	33,33	3	0
2	8	9	12,5	9	12,5	9	12,5
3	10	11	10	10	0	9	10
4	9	10	11,11	10	11,11	9	0
5	12	11	8,33	13	8,33	12	0
6	25	25	0	27	8	27	8
7	11	11	0	12	9,09	12	9,09
8	NA	7		6		6	
9	23	18	21,73	18	21,73	17	26,08
10	NA	6		6		5	
Erro médio			21,88		25,45		21,18

**Fonte: Autoria própria.**

A Figura 29 é referente ao vídeo 8\_01\_R\_130506140000.avi (EMBRAPA, 2013) e teve a quantidade de frames definida em 2880.



**Figura 29 – 2880 frames.**

**Autoria Própria.**

**Tabela 7 – Comparativo entre a contagem automatizada e a manual referente à Figura 29.**

Área	Contagem Humana	Contagem algoritmo	Margem de erro %	Mediana	Margem de erro %	Média	Margem de erro %
Manual	6	5	16,67	5	16,67	4	33,33
1	6	3	50	3	50	3	50
2	11	11	0	10	9,09	10	9,09
3	9	12	33,33	11	22,22	11	22,22
4	14	14	0	13	7,14	13	7,14
5	10	7	30	7	30	6	40
6	29	26	10,34	25	13,79	25	13,79
7	10	12	20	12	20	11	10
8	NA	3		3		2	
9	17	16	5,88	15	11,76	15	11,76
10	NA	5		4		4	
Erro médio			18,46		20,07		21,92

**Fonte: Autoria própria.**

Os resultados das tabelas 2 até 7 foram utilizadas para construir as tabelas de 8 a 11.

A Tabela 8 apresenta o erro médio do algoritmo para cada uma das tabelas de teste, e com todos esses resultados é possível ter a média geral de erro, para cada uma das regiões avaliadas pelo sistema.

**Tabela 8 – Margem de erro do algoritmo por área e por tabela.**

Área	Tabela 2 49 Frames	Tabela 3 49 Frames	Tabela 4 2880 Frames	Tabela 5 7200 Frames	Tabela 6 14400 Frames	Tabela 7 2880 Frames	Erro médio
Manual	25	15,38	25	16,67	100	16,67	33,12
1	42,85	20	33,33	20	33,33	50	33,25
2	12,5	0	12,5	25	12,5	0	10,41
3	9,09	12,5	25	12,5	10	33,33	17,07
4	23,07	11,11	10	10	11,11	0	10,88
5	0	30	13,33	30	8,33	30	18,61
6	0	0	3,33	0	0	10,34	2,27
7	18,18	33,33	9,09	10	0	20	15,1
8	NA	NA	NA	NA	NA	NA	NA
9	6,25	21,73	25	5,26	21,73	5,88	14,30
10	NA	NA	NA	NA	NA	NA	NA

**Fonte: Autoria própria.**

A Tabela 9 apresenta a media de erro das Tabelas 2 a 7 referentes à Mediana, possibilitando assim ter a média geral de erro para a mediana para cada região.

**Tabela 9 – Margem de erro da mediana por área e por tabela.**

Área	Tabela 2 49 Frames	Tabela 3 49 Frames	Tabela 4 2880 Frames	Tabela 5 7200 Frames	Tabela 6 14400 Frames	Tabela 7 2880 Frames	Erro médio
Manual	25	15,38	0	0	125	16,67	30,34
1	28,57	20	33,33	20	33,33	50	30,87
2	12,5	0	12,5	12,5	12,5	9,09	9,84
3	0	0	37,5	25	0	22,22	14,12
4	23,07	11,11	0	0	11,11	7,14	8,73
5	0	20	13,33	30	8,33	30	16,94
6	0	0	6,67	0	8	13,79	4,74
7	18,18	22,22	18,18	30	9,09	20	19,61
8	NA	NA	NA	NA	NA	NA	NA
9	6,25	21,73	29,16	10,52	21,73	11,76	16,85
10	NA	NA	NA	NA	NA	NA	NA

**Fonte: Autoria própria.**

A Tabela 10 apresenta a media geral de erro das Tabelas 2 a 7 referente a média calculada pelo sistema para a quantidade de aves. Na coluna Erro Médio tem-se a média de erro de todas as tabelas por região avaliada.

**Tabela 10 – Margem de erro da média por área e por tabela.**

Área	Tabela 2 49 Frames	Tabela 3 49 Frames	Tabela 4 2880 Frames	Tabela 5 7200 Frames	Tabela 6 14400 Frames	Tabela 7 2880 Frames	Erro médio
Manual	25	15,38	0	16,67	125	33,33	35,89
1	42,85	20	50	40	0	50	33,80
2	0	0	12,5	12,5	12,5	9,09	7,76
3	9,09	0	25	12,5	10	22,22	13,13
4	23,07	0	0	0	0	7,14	5,03
5	0	10	13,33	20	0	40	13,88
6	0	3,44	10	3,57	8	13,79	6,46
7	18,18	22,22	9,09	20	9,09	10	14,76
8	NA	NA	NA	NA	NA	NA	NA
9	0	21,73	29,16	10,52	26,08	11,76	16,54
10	NA	NA	NA	NA	NA	NA	NA

**Fonte: Autoria própria.**

**Tabela 11 – Média do percentual de erro por área e por resultado.**

Área	Algoritmo	Mediana	Media
Manual	33,12	30,34	35,89
R1	33,25	30,87	33,80
R2	10,41	9,84	7,76
R3	17,07	14,12	13,13
R4	10,88	8,73	5,03
R5	18,61	16,94	13,88
R6	2,27	4,74	6,46
R7	15,1	19,61	14,76
R8	NA	NA	NA
R9	14,30	16,85	16,54
R10	NA	NA	NA
Média	17,22	16,89	16,36

**Fonte: Autoria Própria.**

Após avaliação da Tabela 11 constata-se que a área selecionada manualmente apresenta uma margem de erro muito elevada, cuja complexidade para melhorar esse resultado é de grande monta inviabilizando a utilização desse recurso.

Para a área 1 a margem de erro também foi elevada, entretanto o ajuste para essa área é relativamente simples, possibilitando melhor aproveitamento do sistema.

Para as demais áreas analisadas o aproveitamento foi bom, viabilizando o uso do sistema na versão atual.

Comparando a contagem do *frame* com a média, o resultado da média é um pouco melhor. Dos nove resultados avaliados o algoritmo foi um pouco melhor na área selecionada manualmente e nas áreas 1, 6 e 9 enquanto a média foi melhor nas áreas 2, 3, 4, 5 e 7, entretanto onde a média foi melhor a diferença entre ela e o algoritmo superou 5%, e onde o algoritmo foi melhor a diferença entre ele e a média foi inferior a 4%. Interessante observar que cada método teve resultado melhor e, diferentes regiões, o que pode sugerir a análise desses dados em conjunto ao invés de isolados.

Comparando a contagem do algoritmo com a mediana, o resultado do algoritmo foi melhor nas áreas 7 e 9, porém a diferença entre as duas formas não chegou a 4% em nenhuma das áreas.

Desconsiderando as áreas um e a área selecionada manualmente por ter uma margem de erro muito elevada, à média se saiu melhor em seis das sete áreas restantes, porém a diferença entre elas não chegou a 5%.

Após a realização dos testes conclui-se que a quantidade de *frames* ideal para calcular a média e a mediana é abaixo dos 1500 *frames*. A quantidade menor de *frames* é melhor por corresponder a um período de tempo mais próximo do *frame* avaliado, enquanto que uma quantidade muito grande de *frames* vai corresponder a um espaço de tempo muito grande entre o início da avaliação e o momento avaliado manualmente.

## 6 CONSIDERAÇÕES FINAIS

### 6.1 CONCLUSÃO

Para o desenvolvimento desse trabalho foram estudadas ferramentas, algoritmos e técnicas de processamento digital de imagens, com o objetivo de desenvolver um protótipo de sistema capaz de contar a quantidade de aves em cena e verificar a viabilidade de se trabalhar com as imagens atuais, bem como identificar o que pode ser melhorado para uma eventual nova coleta de imagens para esse fim.

Foram descritas as ferramentas utilizadas bem como as técnicas empregadas para o desenvolvimento do sistema, além das dificuldades de algumas dessas técnicas em relação às imagens trabalhadas.

Como resultado tem-se o sistema em si que possibilita contar as aves nos vídeos, com um percentual de erro médio padrão para cada uma das regiões do vídeo. Também sabe-se agora que a forma de aquisição das imagens deve ser feita de cima para baixo em ângulo reto, e deve-se alterar o mínimo possível às câmeras e o ambiente, facilitando o mapeamento dos comedouros e bebedouros que são as principais regiões de interesse dos pesquisadores da avicultura. Ficou resolvida a questão da coloração das imagens geradas com câmeras infravermelhas, deixando de ser um complicador, uma vez que nesse trabalho foi identificada uma forma de contornar esse problema através do pré-processamento das imagens, que apresentou ótimos resultados identificando uma forma de corrigir o brilho da cena e separar os objetos de interesse do fundo da imagem, criando um modelo e subtraindo as imagens para poder segmentá-la.

Este projeto proporcionou a aquisição de muitos conhecimentos ao seu executor em uma área que lhe era totalmente desconhecida e de complexidade elevada. A área de processamento digital de imagens é empolgante e desafiadora, instigando a necessidade de estudar e se atualizar para acompanhar os avanços tecnológicos que ela proporciona.

### 6.2 TRABALHOS FUTUROS

Analisar a textura das imagens para tentar distinguir as aves umas das outras.

Replicar o trabalho com imagens coletadas para esse fim.

Utilizar técnicas de rastreamento de objetos para monitorar as aves.

## REFERÊNCIAS BIBLIOGRÁFICAS

ABREU, PAULO GIOVANNI. **REFERENCIAL TEÓRICO CONCÓRDIA SC: EMBRAPA SUÍNOS E AVES**, 2013.

ALBUQUERQUE, Márcio e Albuquerque, Marcelo. **Processamento de imagens Métodos e Análises**. CBPF/MCT Disponível em: <  
<http://www.cbpf.br/cat/pdsi/pdf/ProcessamentoImagens.PDF>> Acesso em 23/08/2014

BARBEDO Jayme Garcia Arnal. **Estado da arte das técnicas de contagem de elementos específicos em imagens digitais**, 1º Ed, Embrapa informática agropecuária, Campinas SP, 2012. Disponível em <  
<http://www.infoteca.cnptia.embrapa.br/bitstream/doc/930751/1/documento120.pdf> > Acesso em 22/08/14

CODEBLOCKS. **Official Wiki Code Blocks**. Disponível em <  
[http://wiki.codeblocks.org/index.php?title=Main\\_Page](http://wiki.codeblocks.org/index.php?title=Main_Page)> Acesso em: 23/08/2014

EMBRAPA, **Vídeos do interior de um aviário**. 2013 243 vídeos com extensão avi, preto e branco 4,63 cm X 2,60 cm.

FACON, Jacques. **Morfologia Matemática: Teoria e exemplos**. Ed. Curitiba: Editora Universitária Champagnat da Pontifícia Universidade Católica do Paraná, 1996. (FACON, 1996)

GOMES, Otávio. **Processamento e análise de imagens aplicados à caracterização automática de matérias** PUCRJ 2001. Disponível em: < <http://www.dcm.puc-rio.br/cursos/ipdi/html/ogomesmestrado.pdf>> Acesso em 23/08/14

GONZALEZ, Rafael; WOODS, Richard E. **Processamento Digital de Imagens** 3º edição. Ed. São Paulo: Pearson Prentice Hall, 2010. (GONZALEZ, WOODS 2010)

ITSEEZ 2014. **Sobre OpenCV** Disponível em: < <http://opencv.org/about.html>> Acesso em: 23/08/2014

MACHADO, Jonathan D. **O que são frames por segundo?** Disponível em <<http://www.tecmundo.com.br/video/10926-o-que-sao-frames-por-segundo-.htm>> Acesso em 12/11/2014.

MARQUES FILHO; VIEIRA NETO, Hugo. **Processamento Digital de Imagens**. Rio de Janeiro: Brasport, 1999.

MOGAMI, Cristina Akami. **Desenvolvimento de metodologias para determinação do bem estar e massa corporal de frangos de corte por meio de análise digital de imagens**. Universidade Federal de Viçosa 2009. Disponível em: <<http://www.ufv.br/dea/ambiagro/gallery/publica%C3%A7%C3%B5es/tesecristinads.pdf>> Acesso em: 26/08/2014

MUCELIN, Carlos Alberto. **Estatística Elementar e Experimental Aplicada às Tecnologias**. Medianeira CEFET-PR, 2003.

PAULA FILHO, PEDRO LUIZ. **Processamento de Imagens Aplicado à Agroindústria**. Operações com imagens. Do curso de Pós Graduação Aplicada à Agroindústria UTFPR 2014

PEDRINI, Hélio; SCHWARTZ, Willian Robson. **Análise de imagens Digitais: Princípios, Algoritmos e Aplicações**. Ed. São Paulo: Thomson Learning, 2008.

PITERI, Marco Antonio; RODRIGUES, José Carlos. **Fundamentos de Visão Computacional**. Ed. Presidente Prudente: FCT/UNESP-PP, 2011. (PITERI, RODRIGUES 2011)

PORTELA, Nara Miranda. 2009. **Contagem automática de ovos de Aedes Aegypti em imagens de ovitrampas**. Disponível em: <[http://www.dominiopublico.gov.br/pesquisa/DetalheObraForm.do?select\\_action=&co\\_obra=167757](http://www.dominiopublico.gov.br/pesquisa/DetalheObraForm.do?select_action=&co_obra=167757)> Acesso em: 06/10/2014.

SALIS, Turibio Tanu, PEREIRA, Guilherme Augusto Silva. 2007. **Contagem automática de tarugos de aço por meio de visão computacional**. Disponível em: <<http://www.ppgee.ufmg.br/documentos/PublicacoesDefesas/786/ContagemAutomatica.pdf>> Acesso em 03/10/2014. (SALIS, PEREIRA, 2007)

VEIT, Eliane Angela. ARAUJO, Ives Solano. **Laboratório didático de física**. Analisando imagens e vídeos com o computador. Disponível em <<http://www.if.ufrgs.br/cref/uab/lab/tracker.html#entendendo>> Acesso em 12/11/2014.



### ANEXO A – MODELO DO ARQUIVO GERADO PELO SISTEMA.

A primeira coluna de nome *frame* representa o ultimo *frame* calculado, as colunas que iniciam com a letra “R” e seguidas de um número, identificam a área calculada. Na sequencia de cada identificação de área tem as colunas quantidade, mediana e média. Por fim as três ultimas colunas são os resultados para a área selecionada manualmente.

A Figura 30 é uma imagem do arquivo com todas as colunas.

The image shows a full spreadsheet with 26 columns and 8 rows. The first row contains headers for each column. The first column is 'Frame', the second is 'R1', and so on, up to 'R26'. The data rows follow a consistent pattern: Frame, R1, Quantity, Median, Mean, R2, Quantity, Median, Mean, R3, Quantity, Median, Mean, R4, Quantity, Median, Mean, R5, Quantity, Median, Mean, R6, Quantity, Median, Mean, R7, Quantity, Median, Mean, R8.

Figura 30 – Imagem do arquivo com todas as colunas.

Fonte: Autoria Própria.

A Figura 31 é uma imagem de parte do arquivo para que os dados contidos nele possam ser melhor visualizados.

	A	B	C	D	E	F	G	H	I	J
1	Frame	R1	Quantidade	Mediana	Media	R2	Quantidade	Mediana	Media	R3
2	200	R1	4	4	4	R2	8	9	8	R3
3	400	R1	3	4	3	R2	9	9	8	R3
4	600	R1	4	4	3	R2	9	9	9	R3
5	800	R1	4	4	4	R2	9	9	9	R3
6	1000	R1	4	4	3	R2	9	9	8	R3
7	1200	R1	4	4	3	R2	9	9	8	R3
8	1400	R1	4	4	3	R2	8	9	8	R3

Figura 31 – Imagem de parte do arquivo gerado.

Fonte: Autoria Própria.