



Universidade Católica Dom Bosco

Curso de Bacharelado em Engenharia de Computação

**Desenvolvimento de um sistema de identificação de
espécies de peixes utilizando Visão Computacional e
aprendizagem automática**

Rafael Sanches Telles

Prof. Dr. Hemerson Pistori

*Relatório Final apresentado à Coordenação do
Curso de Bacharelado em Engenharia de Compu-
tação da Universidade Católica Dom Bosco como
parte dos requisitos para a obtenção do título de Ba-
charel em Engenharia de Computação.*

UCDB - Campo Grande/MS - 06/2013

a minha família e amigos.

Agradecimentos

Primeiramente agradeço a Deus que iluminou o meu caminho durante toda essa longa caminhada. Agradeço também a toda minha família, que em momento algum deixou de acreditar em meu potencial e sempre me incentivou a ir em busca dos meus sonhos. Em especial ao meu avô, que por motivo de força maior não conseguiu acompanhar até o final essa conquista minha. Agradeço também a todos meus amigos, que sempre foram apoiadores dos meus sonhos e assim como meus familiares, em momento nenhum deixaram de acreditar em meu potencial.

Agradeço aos ensinamentos do meu orientador Dr. Hemerson Pistori que compartilhou comigo todos os seus conhecimentos durante o tempo que participei do grupo INOVISAO. Agradeço também, pela rigidez que sempre teve durante o desenvolvimento desse trabalho o que contribuiu para que ao final do trabalho pudesse eu me tornasse uma pessoa mais capacitada e muito mais forte.

Agradeço também o empresa TecSinapse, que desde o início, sempre me apoiou no desenvolvimento desse trabalho e contribuiu muito para que o trabalho pudesse ser concluído com sucesso. Agradeço também a empresa Flor de Liz, que gentilmente cedeu a oportunidade de construir o banco de imagens desse trabalho.

Resumo

Este trabalho tem como principal objetivo o desenvolvimento de uma aplicação para reconhecimento de espécies de peixes. A aplicação tem como principal objetivo realizar o reconhecimento de algumas espécies de peixes utilizando a câmera (*webcam*) ou imagens já disponíveis em disco, utilizando os algoritmos SURF e SIFT para detectar e descrever os pontos de interesse. Posteriormente foram aplicadas as técnicas para casamento de modelos de Força Bruta e FLANN. Para cada espécie foram capturadas cerca 30 imagens em diferentes ângulos e distâncias, assim constituindo um banco de imagens com uma grande diversidade de imagens. O protótipo foi desenvolvido na linguagem C, utilizando a biblioteca de Visão Computacional OpenCV. Para cada combinação (detecção/descrição + casamento de modelos) foram realizadas três repetições da execução. Para cada repetição foram obtidos os resultados referentes as métricas taxa de verdadeiros positivos, taxa de falsos positivos, medida-F, acurácia e precisão. Ao final do experimento foi possível observar que o algoritmo SIFT com a técnica para casamento de modelos de Força Bruta obteve o melhor resultado em todas as métricas, com exceção da métrica referente a taxa de falsos positivos que obteve um resultado igual a combinação SURF com FLANN.

Palavras-chave: visão computacional, reconhecimento de padrões, SURF, SIFT, casamento de modelo e peixes.

Sumário

Lista de Figuras

Lista de Tabelas

1	Introdução	p. 12
2	Trabalhos Correlatos	p. 16
2.1	SURF e SIFT	p. 16
2.2	Classificação de espécies de peixes	p. 17
3	Fundamentação Teórica	p. 20
3.1	Visão Computacional	p. 20
3.2	Reconhecimento de padrões	p. 21
3.3	Validação dos resultados	p. 22
3.4	Teste de Friedman	p. 28
3.5	Análise pós-teste	p. 28
4	Principais espécies ornamentais da cidade de Campo Grande - MS	p. 29
5	Técnicas de Identificação de pontos de interesse	p. 34
5.1	SURF: Speeded-Up Robust Features	p. 35
5.1.1	Detecção de pontos de interesse	p. 35
5.1.1.1	Imagens Integrais	p. 36
5.1.1.2	Matriz Hessiana baseada nos pontos de interesse . . .	p. 36
5.1.1.3	Pirâmide de escala e oitavas	p. 38

5.1.1.4	Supressão de não máximos em 3D	p. 39
5.1.2	Descrição dos pontos de interesse	p. 41
5.1.2.1	Cálculo de orientação dominante para invariância à rotação	p. 41
5.1.2.2	Distribuição de respostas de Haar Wavelets	p. 42
5.1.2.3	Vetor de 64 atributos para cada ponto de interesse . .	p. 43
5.2	SIFT: Scale-invariant Feature Detector	p. 43
5.2.1	Detecção do espaço de escala	p. 44
5.2.2	Localização dos pontos de interesse	p. 46
5.2.3	Atribuição da orientação	p. 46
5.2.4	Descrição dos pontos chaves	p. 47
6	Casamento de Modelos	p. 50
6.1	FLANN	p. 51
6.2	Força Bruta	p. 52
7	Metodologia	p. 53
7.1	Banco de imagens	p. 53
7.2	Delineamento experimental	p. 55
8	Resultados, Análise e Discussão	p. 59
9	Conclusões e Trabalhos Futuros	p. 66
	Referências	p. 68

Lista de Figuras

1	Projeto arquitetônico do Aquário do Pantanal.	p. 13
2	Inspeção de peixes em uma indústria de pescado no Chile.	p. 13
3	Matriz de confusão gerada a partir de um conjunto de instâncias	p. 23
4	Matriz de confusão gerada a partir de um conjunto de instâncias	p. 24
5	Matriz de confusão gerada a partir de um conjunto de instâncias	p. 24
6	Resultado da matriz de confusão para um problema contendo 5 classes	p. 27
7	Espécie de peixe Beta.	p. 30
8	Espécie de peixe Platy.	p. 30
9	Espécie de peixe Kinguio.	p. 31
10	Espécie de peixe Guppy.	p. 32
11	Espécie de peixe Tricogaster azul.	p. 32
12	Espécie de peixe Acará Bandeira.	p. 33
13	Espécie de peixe Tricogaster azul.	p. 33
14	Peixe da espécie platy sangue. Os círculos existentes em várias áreas da figura, mostram os pontos de interesse.	p. 34
15	Utilizando imagens integrais, são necessárias apenas três adições e quatro acessos à memória para calcular a soma de intensidades dentro de uma região retangular de qualquer tamanho.	p. 36
16	Matriz Hessiana. x representa um ponto da imagem e σ determina a escala. L_{xx} derivada parcial no eixo x , L_{yy} derivada parcial no eixo y e L_{xy} derivada parcial no eixo xy	p. 37
17	Da esquerda para a direita: Derivadas parciais de segunda ordem na direção x e xy e suas aproximações, utilizando filtros de caixa. As regiões em cinza são iguais a zero.	p. 38

18	Fórmula utilizada para realizar a aproximação do determinante. D_{xx} derivada parcial no eixo x, D_{yy} derivada parcial no eixo y e D_{xy} derivada parcial no eixo xy.	p. 38
19	Em vez de utilizar a redução do tamanho da imagem (esquerda), a utilização de imagens integrais permite o aumento de escala do filtro por um custo constante (direita).	p. 38
20	Exemplo de filtros em escalas diferentes, 9x9 (esquerda) e 15x15 (direita)	p. 39
21	Representação da escala de oitavas. O eixo horizontal representa o logarítmico das escalas. O eixo vertical representa as oitavas.	p. 40
22	Exemplo real de detecção de pontos de interesse	p. 40
23	Representação do descritor de pontos de interesse. Note que os pontos interessantes estão marcados com um quadrado e a linha reta que vai do centro a algum lado é a direção onde ocorre maior variação.	p. 41
24	Filtros em x (esquerda) e filtros em y (direita)	p. 41
25	Gráfico em formato de pizza que representa a resposta, após aplicar os filtros. Tamanho da pizza é fixo.	p. 42
26	Representação do quadrado de respostas de Wavelet	p. 43
27	Exemplo geração de vetor de atributos para cada ponto de interesse . .	p. 44
28	Exemplo de aplicação do filtro para a detecção do espaço de escala. Esquerda imagem original. Centro filtro Gaussiano. Direta imagem resultante do filtro da diferença de Gaussianos.	p. 45
29	Representação utilizada por Lowe (2004) para representar como é obtido as DoG para diversas oitavas de uma determinanda imagem.	p. 46
30	Mostra as etapas da seleção dos pontos de interesse. (a) imagem original 233x189. (b) os 832 pontos de interesse locais em máximos e mínimos da função DoG. (c) após a aplicação do limiar de contraste, restam 729 pontos de interesse. (d) após a retirada dos pontos que não interessam (possuem baixo contraste ou são sensíveis a ruídos) restam 536 pontos de interesse. Cada ponto de interesse é representado por uma flexa, que indica a direção, sentido e escala.	p. 47
31	Histograma de orientações para cada pontos de interesses.	p. 48

32	Exemplo de um descritor para um ponto de interesse.	p. 48
33	Construção de pontos de interesse proposto por Lowe (2004). Descritor 2x2 utilizando 48 elementos.	p. 49
34	(a) e (b) espécie kinguio. (c) e (d) espécie acará bandeira. (e) e (f) espécie molinésia preta. (g) platy sangue. (h) e (i) tricogaster azul. De (a) até (h) corresponde ao conjunto de treinamento e (i) seria a imagem de “entrada”. (a) e (f) nenhum ponto de interesse detectado. (b) (d) (e) e (g) foi detectado 1 ponto de interesse. (c) dois pontos de interesses detectados. (h) cinco pontos de interesse detectados. (i) onze pontos de interesse detectados	p. 51
35	Ambiente de Captura	p. 54
36	Exemplo de imagem capturada no seu formato original.	p. 55
37	Cinco espécies de peixes que foram utilizadas no banco de imagens: Tricogaster Azul(a), Acará Bandeira(b), Kinguio(c), Platy Sangue(d) e Molinésia Preto(e)	p. 56
38	Grade contendo todas as espécie do banco de imagens em diferentes ângulos. Linha 1: Acará bandeira. Linha 2: Kinguio. Linha 3: Molinésia preta. Linha 4: Platy sangue. Linha 5: Tricogaster azul.	p. 57
39	Exemplo de imagem para realizar o reconhecimento. Espécie de peixe Platy sangue. Imagem disponível no banco de imagens.	p. 58
40	Resultado da análise <i>post hoc</i> para a métrica taxa de verdadeiros positivos	p. 60
41	Resultado da análise <i>post hoc</i> para a métrica taxa de falsos positivos . .	p. 60
42	Resultado da análise <i>post hoc</i> para a métrica medida-F	p. 61
43	Resultado da análise <i>post hoc</i> para a métrica acurácia	p. 62
44	Resultado da análise <i>post hoc</i> para a métrica precisão	p. 63
45	Peixe da espécie kinguio. (a) Imagem original e (b) imagem em tons de cinza	p. 64

- 46 Espécie de peixe platy sangue. (a) representa a imagem que está sendo utilizada como entrada, e (b) a imagem comparada exibindo os pontos de interesse. A reta traçada entre os pontos de interesse de imagens distintas representa os pontos interessantes semelhantes nas duas imagens. p.64
- 47 Espécie de peixe tricogaster. (a) representa a imagem que está sendo utilizada como entrada, e (b) a imagem comparada exibindo os pontos de interesse. A reta traçada entre os pontos de interesse de imagens distintas representa os pontos semelhantes nas duas imagens. p.65

Lista de Tabelas

1	Taxa de verdadeiros positivos	p. 59
2	Taxa de falsos positivos	p. 59
3	Medida-F	p. 61
4	Acurácia	p. 62
5	Precisão	p. 62

1 *Introdução*

O conjunto de métodos e técnicas através dos quais os sistemas computacionais interpretam as imagens é definido como Visão Computacional, uma área da computação muito promissora, que pretende tornar possível que os sistemas computacionais possam compreender cores e formas a sua volta (RIOS, 2010). Embora algumas áreas já sejam beneficiadas com os estudos de métodos que ajudam na identificação de fenômenos e eventos, como exemplo da medicina, astronomia, robótica, em outras há muito o que evoluir.

A Visão Computacional e os seres humanos estão cada vez mais próximos, possibilitando a criação de aplicações para o reconhecimento de padrões, que antes acreditava ser uma tarefa impossível. No entanto, como é descrito por Rios (2010), a identificação de padrões utilizando técnicas de Visão Computacional não é uma tarefa trivial, porém já existem algumas bibliotecas que auxiliam no desenvolvimento desta tarefa, um exemplo delas é a OpenCV.

No estado do Mato Grosso do Sul mais especificamente na cidade de Campo Grande está sendo construído um aquário, chamado de Aquário do Pantanal. Esse aquário será o maior aquário de água doce do mundo, terá 24 tanques com aproximadamente 6,5 milhões de litro de água e cerca de 260 espécies diferentes¹. Com isso é possível observar a possibilidade de mercado com o uso de uma aplicação que auxilie no reconhecimento de espécies de peixes, onde seja possível uma pessoa identificar qual espécie de peixe que está sendo visualizada, além de exibir informações adicionais sobre a espécie, como por exemplo, onde é mais comum encontrar tais espécie, comprimento aproximado, família que pertence, tamanho máximo, classe que pertence, sua origem, peso aproximado e informações corporais. A Figura 1 mostra como ficará o aquário quando acabado.

¹<http://www.pantanal-brasil.com/page.aspx?pagina=232>



Figura 1: Projeto arquitetônico do Aquário do Pantanal.

Segundo Rodrigues (2009), o problema da classificação de espécies de peixes também é um problema recorrente na indústria. A Figura 2 ilustra um problema muito comum na indústria do pescado, onde os peixes chegam por meio de um caminhão e são deixados nas esteiras e posteriormente são catalogadas as informações referentes a tamanho, peso, quantidade e espécie (RODRIGUES, 2009). Essa classificação é feita por seres humanos que muitas vezes podem estar cansados prejudicando a classificação.

Dispositivos eletrônicos poderiam realizar esta classificação de forma mais ágil, sendo possível diminuir a probabilidade de erros e consequentemente aumentar os resultados da empresa. Outro fator que é possível observar é que com a diminuição de trabalhadores no setor de catalogação poderia, por exemplo, aumentar o número de pessoas que realizam a busca pelo pescado.



Figura 2: Inspeção de peixes em uma indústria de pescado no Chile.

Conforme Nery et al. (2005), a classificação de peixes não é uma tarefa fácil, já que há várias características que devem ser avaliadas, por exemplo, coloração, largura, comprimento e uma série de outras características que são importantes para se ter uma classificação precisa. Ainda segundo Nery et al. (2005), existem 47 características diferentes e importantes para cada espécie para que se consiga um nível bom de exatidão na identificação.

O trabalho de visualizar as diferentes espécies de peixes em um determinado aquário é grande, porque além de exigir um conhecimento sobre as espécies, os peixes se movimentam de forma rápida o que dificulta a identificação da espécie (NERY et al., 2005). Por esses fatos, a identificação de diferentes espécies, em diferentes aquários, permanece sendo um desafio.

Alguns estudos na área apontam formas eficientes de classificação através de um sistema automático com exatidão de até 92% (RODRIGUES, 2009). Contudo, como mostrado por Rodrigues (2009), é necessária a construção de uma base de dados que permita o desenvolvimento e a validação do sistema de classificação.

O desenvolvimento de uma aplicação foi o principal objetivo deste projeto, que visa realizar o reconhecimento de algumas espécies de peixes. Para realizar este reconhecimento foi feito um levantamento sobre as principais espécies de peixes do estado de Mato Grosso do Sul, para que o estudo fosse focado nestas espécies. Também foi construído um banco de imagens com cinco classes diferentes. Cada classe possui cerca de 30 imagens em diferentes ângulos. Para desenvolver a aplicação foi utilizada a biblioteca de Visão Computacional OpenCV, que é escrita na linguagem C. Para validar os resultados foram utilizadas as métricas taxa de verdadeiros positivos, taxa de falsos positivos, medida-F, acurácia e precisão.

No Capítulo 2 é apresentado um pouco sobre os trabalhos correlatos da área. Na primeira sessão serão apresentados os trabalhos que envolvem os algoritmos SURF e SIFT para, de alguma forma, realizar o reconhecimento de padrões; na segunda sessão são apresentados os trabalhos que realizam o reconhecimento de espécies de peixes utilizando alguma outra técnica. O Capítulo 3 refere-se à fundamentação teórica, apresentando alguns conceitos importantes sobre o trabalho. O capítulo 4 mostra uma pesquisa realizada sobre as principais espécies do estado de Mato Grosso do Sul, bem como uma descrição sobre cada espécie. Já no Capítulo 5 é explicado o funcionamento dos algoritmos SURF e SIFT, que

são utilizados para reconhecimento de padrões. No próximo Capítulo são apresentadas algumas formas de realizar o casamento de modelos. A forma como foi realizado todo o processo de captura das imagens, desenvolvimento e obtenção dos resultados é mostrada no Capítulo 7. Os resultados, análises e discussões estão descritos no Capítulo 8, por fim, o Capítulo 9 apresenta as conclusões e os trabalhos futuros.

2 *Trabalhos Correlatos*

Este capítulo descreve os trabalhos correlatos que foram encontrados na literatura, separados em duas seções. A primeira sessão descreve alguns trabalhos que utilizam as mesmas técnicas utilizadas nesse trabalho para identificação de pontos de interesse, bem como os resultados obtidos. A segunda sessão é referente aos trabalhos que tentaram resolver o problema da classificação de espécies de peixes utilizando diversas técnicas.

2.1 SURF e SIFT

O trabalho de Rachid e Pereira (2009) realiza um estudo de duas técnicas utilizadas para o reconhecimento de padrões (SURF-SIFT) em dispositivos móveis. Nesse estudo, foi realizada uma comparação voltada para o problema de reconhecimento de imagens, comparando resultados obtidos por autores anteriormente. Neste trabalho, o algoritmo SIFT teve um melhor desempenho, porém um custo computacional maior se comparado ao algoritmo SURF.

Morais (2011) propõe o desenvolvimento de uma sistema de realidade aumentada utilizando o algoritmo SURF para realizar a busca de marcadores. O sistema proposto por Moraes (2011) consiste em encontrar um determinado marcador, previamente fornecido ao sistema. Para encontrar o marcador, segundo Moraes (2011), é necessário realizar as seguintes etapas: calibração da câmera, adicionar ao sistema o marcador que deseja encontrar, realizar a busca de pontos de interesse no marcador utilizando o algoritmo SURF, buscar o marcador no *frame* utilizando também o algoritmo SURF e por fim, caso encontre o marcador, estimar a posição e realizar uma projeção do marcador. O sistema foi desenvolvido inicialmente para *desktops* e posteriormente migrou-se para plataforma móvel *android*. O autor ainda cita que a aplicação para dispositivos móveis obteve um baixo desempenho, onde cada detecção demora cerca de 15 segundos não sendo possível comparar com sistemas já desenvolvidos.

Holanda et al. (2010) utiliza o algoritmo SURF para realizar as medições de distâncias entre dois robôs, utilizando juntamente os algoritmos clássicos de visão em estéreo no desenvolvimento de sistemas de Visão Computacional em 3D. O sistema proposto por Holanda et al. (2010) é composto por duas *webcams* para representar a visão do robô. Um dos principais objetivos do trabalho seria encontrar pontos correspondentes em ambas *webcams*. Cada *webcam* é responsável por realizar a capatura, um pré-processamento da imagem capturada e aplicar o algoritmo SURF, para detectar os pontos de interesse. Ao final dessa primeira parte nas duas *webcams*, realiza uma comparação para verificar se há pontos semelhantes nas duas imagens. Caso encontre pontos interessantes nas duas imagens, realiza uma série de cálculos para verificar a distância. Neste trabalho o algoritmo SURF obteve um ótimo desempenho tendo um erro percentual menor que 10% nos testes.

O trabalho de Oliveira e Rocha (2010) propôs um comparativo entre duas técnicas (SIFT e SURF) para encontrar pontos de interesse em ambientes subaquáticos. Ainda segundo Oliveira e Rocha (2010), os dados utilizados nos testes foram obtidos através da navegação de robôs subaquático com diversos tipos de distorções, possuindo assim diferentes tipos de imagens com diversos tipos de ruídos. Os algoritmos SURF e SIFT realizavam a detecção dos pontos de interesse em tempo real, utilizando a *webcam* do computador. O sistema proposto por Oliveira e Rocha (2010) é um sistema para funcionar em tempo real. Ao final do trabalho, o autor comparou as duas técnicas. Um comparativo muito interessante, já o cenário abordado por este autor é muito semelhante ao estudado nesse trabalho. Neste trabalho o algoritmo SIFT apresentou melhores resultados.

2.2 Classificação de espécies de peixes

Rodrigues (2009) busca resolver o problema da classificação automática de espécies de peixes utilizando Sistemas Imunológicos Artificiais e Visão Computacional. Neste trabalho são utilizadas 9 espécies de peixes, são elas: Canivete (*Apareiodon piracicabae*), Barbado (*Pinirampus pirinampu*), Cará (*Cichlasoma fasciatum*), Carpa (*Cyprinus carpio*), Cascudo (*Rinelepis aspera*), Lambari (*Astyanax scabripinnis*), Pacu (*Piractus mesopotamicus*), Surubim (*Steindachneridion scripta*) e Tucunaré (*Cichla monoculus*). Os experimentos foram realizados com dois grupos.

O primeiro grupo de imagens a ser testado foi o de peixes conservados em formol, con-

tendo 6 espécies, cujo experimento apresentou um nível de exatidão próximo a 86%. O segundo grupo de imagens testado foi o de imagens subaquáticas e foi testado a capacidade de identificar 4 espécies. A classificação realizada por meio da técnica citada acima possibilitou a identificação visual de objetos utilizando informações úteis como forma, cor e textura (RODRIGUES, 2009). Para esse segundo grupo o nível de exatidão chegou próximo a 90%.

Segundo Alsmadi et al. (2010), o problema de reconhecimento de peixes pode ser resolvido por meio de extrações robustas das características de tamanho e forma utilizando Rede Neural. Este estudo aplicou o reconhecimento para 20 diferentes espécies de peixes com diferentes formas e tamanhos, construindo um banco de imagens total de 350 imagens. No estudo realizado por Alsmadi et al. (2010), o principal objetivo foi reconhecer um padrão isolado de interesse com base na extração robusta de características. O classificador proposto teve um ótimo desempenho atingindo a precisão de 86%.

Lee et al. (2004) realiza uma extensão do problema, tentando identificar espécies de peixes e armazenar tal informação, para que seja feito um controle de migração das espécies. O sistema completo funciona na seguinte ordem: aquisição de imagens de peixes de forma automática, extração do contorno do mesmo, categorização do peixe e armazenamento dos dados obtidos. Neste trabalho, não há informações sobre o número de classes que foi utilizado. Conforme é descrito no trabalho, o reconhecimento com base nos contornos tem mostrado resultados satisfatórios, porém a descrição desses pontos por vezes, não pode ser obtido com muita precisão.

Chambah et al. (2004) apresenta uma melhoria no problema de classificar as espécies em tempo real. O método de correção proposto é baseado no modelo ACE (*Automatic Color Equalization*), um algoritmo de equalização não supervisionada de cor. O método ACE é uma abordagem perceptiva baseada em alguns mecanismos de adaptação do sistema visual humano, em especial leveza e constância de cores. Segundo Chambah et al. (2004), este tipo de abordagem apresenta várias vantagens, entre elas estão: não é supervisionada, o sistema é robusto e apresenta propriedades locais de filmagem que levam a resultados mais eficazes. Os resultados apresentados ao final deste trabalho foram preliminares, porém segundo o autor, foram satisfatórios.

É apresentada por Nery et al. (2005), uma metodologia de classificação de peixes baseada

em uma técnica robusta de seleção de recurso. Ao contrário da maioria dos trabalhos existentes para classificação de peixes que propõem um descritor e não analisa os seus impactos individuais na tarefa de classificação geral, nesse trabalho é proposto um conjunto geral de características e seus pesos correspondentes, que devem ser utilizadas como uma informação à *priori* pelo classificador.

No entanto, segundo Nery et al. (2005), em vez de “perder” tempo estudando técnicas para melhorar a estrutura de classificadores, que neste caso é considerada como uma “caixa preta”, assim concentrando a pesquisa na determinação de que as informações de entrada devem trazer uma discriminação de peixe robusta (NERY et al., 2005). É proposto um conjunto geral de 20 características para ser utilizadas como priori pelo classificador. Tais como, tamanho, forma, coloração e textura, são características que são avaliadas e para cada uma destas característica é determinado um peso em relação ao classificador. Os resultados obtidos foram satisfatórios, e o índice de reconhecimento ficou em torno de 75%.

3 *Fundamentação Teórica*

3.1 Visão Computacional

Segundo Yang (2007), é muito difícil criar sistemas de Visão Computacional que correspondam às capacidades cognitivas dos seres humanos, bem como identificar por exemplo, qual espécie de peixe está sendo observada. De acordo com Yang (2007), as principais razões encontradas para tal dificuldade podem referir-se aos seguintes fatos: a constante variação de iluminação, a dificuldade de generalizar objetos a partir de um conjunto pré-definido de imagens e a posição relativa de um objeto em relação à câmera.

Conforme aponta Riesenhuber e Poggio (2000) para reconhecer uma imagem qualquer, é preciso separar este reconhecimento em duas tarefas principais: identificação e categorização. A identificação consiste em a partir de uma determinada imagem (imagem de ‘entrada’) extrair características para que possa ser utilizada na etapa posterior (categorização). A detecção consiste em categorizar a imagem de ‘entrada’ de acordo com um conjunto de imagens existente. Ainda segundo Riesenhuber e Poggio (2000), existem algumas técnicas de Visão Computacional relativamente fáceis para a identificação dos objetos. Já a categorização é uma tarefa um pouco mais complicada tendo em vista que exige um certo conhecimento do objeto que está sendo identificado.

De acordo com Forsyth e Ponce (2003), para realizar o reconhecimento de tal objeto utilizando a Visão Computacional é necessária a captura da imagem, seguida do pré-processamento da mesma, visando reduzir os ruídos e as imperfeições contidas na imagem e que podem atrapalhar as etapas seguintes. Após esse pré-processamento a mesma passa por um processo chamado segmentação, que consiste em separar a imagem em regiões que satisfazem um critério de uniformidade, ou seja, separar em objetos úteis para que sejam processados posteriormente. Para classificar o objeto, utiliza-se reconhecimento de padrões explicado a seguir.

Em sistemas de Visão Computacional, geralmente é utilizado imagens digitais. Imagens digitais são imagens bidimensionais que utilizam números binários codificados, assim sendo possível realizar diversas operações. Essas operações podem ser de transferência, reprodução, armazenamento ou até mesmo um processamento da mesma por meios eletrônicos (SCURI, 1999).

3.2 Reconhecimento de padrões

O Reconhecimento de objetos é uma das habilidades mais encantadoras que os seres humanos possuem desde a infância. Com um simples olhar um ser humano é capaz de detectar um objeto, além de determinar sua forma, cor e muitas outras características independentemente da sua variação de aparência, mudança de postura e iluminação (GRACIANO, 2007).

O reconhecimento de padrões em Visão Computacional envolve técnicas de aprendizagem de máquinas e, através de exemplos, classifica os padrões em suas classes de forma automática. Segundo Graciano (2007), o reconhecimento de padrões trata-se da extração e da classificação das características dos dados embasado em amostras padrões. Geralmente um sistema de reconhecimento de padrões é constituído basicamente pela obtenção dos dados e seleção dos objetos a serem classificados com as características procuradas. Entre as diversas técnicas existentes para realizar a extração das características está os algoritmos SURF e SIFT, que serão explicados detalhadamente no Capítulo 5.

Conforme Brunelli (2009), o casamento de modelos é um conjunto de técnicas que quantificam a semelhança existente entre duas imagens digitais, ou até, uma parte delas. Ainda segundo Brunelli (2009), as imagens digitais são representadas com matrizes numéricas - imagens monocromáticas demandam uma única matriz e imagens coloridas requerem várias matrizes, sendo uma por componente de cor¹. Alguns autores utilizam o termo *Template Matching*, que não difere no conceito, apenas na nomenclatura.

Segundo Santos (2011), o casamento de modelos consiste em mover a imagem modelo por todas as posições possíveis na imagem onde se procura por esse modelo e calcular um índice numérico que indica quão bem o modelo ajusta a imagem nessa posição. Esta

¹Os componentes de cores são: R, G e B

correspondência é feita *pixel a pixel*. O funcionamento detalhado da técnica de casamento de modelos será explicado no Capítulo 6.

3.3 Validação dos resultados

Uma das etapas importantes no desenvolvimento de um trabalho de Visão Computacional é a parte de validação dos resultados. Essa etapa consiste em analisar os resultados obtidos através de algumas métricas pré-estabelecidas. Na área de Visão Computacional, há diversas métricas que avaliam diferentes tipos de informações. As métricas são escolhidas pelo desenvolvedor (ou pesquisador) de acordo com o problema que sendo desenvolvido. A fim de avaliar os resultados obtidos, alguns testes de hipóteses serão aplicados para esse problema, por exemplo, teste de Friedman. Para este trabalho foram escolhidas as métricas taxa de verdadeiros positivos, taxa de falsos positivos, medida-F, acurácia e precisão. Foram escolhidas essas métricas pois têm sido utilizadas com frequência na análise de problemas similares. Todas as métricas serão explicadas a seguir.

Considere que tenhamos o problema de classificar uma determinada imagem, contendo o rosto de alguma pessoa, como triste ou alegre. Para esse problema o resultado do classificador poderá assumir um dos quatro valores, VP - Verdadeiros Positivos, FP - Falsos Positivos, FN - Falsos Negativos e VN - Verdadeiros Negativos. Para essa situação, tendo em mão um classificador e a imagem que deseja classificar é possível ter quatro situações distintas. Se a imagem que deseja classificar é verdadeira, ou seja, tem um rosto alegre, e é classificada como alegre, então o resultado dessa classificação é um verdadeiro positivo, pois o resultado do classificador é o resultado real da imagem. Caso essa mesma imagem seja classificada com falsa, ou seja, como triste, então ela é um falso negativo. Considere agora que tenhamos uma imagem com o rosto triste, e essa imagem desejamos classificar. Se essa imagem é classificada como triste, então ela é um verdadeiro negativo, ou seja, o resultado da classificação é o mesmo da imagem. Caso seja classificada como verdadeira, ou seja, como se tivesse um rosto alegre, então é um falso positivo. Abaixo é exibida a matriz de confusão (Figura 3), que tem como principal objetivo verificar se para um determinado problema o valor esperado foi igual (ou próximo) ao valor real. A Figura 3 mostra a matriz de confusão para problemas de duas classes.

		Classe Verdadeira	
		p	n
Classificada Como	p	Verdeiro Positivo	Falso Positivo
	n	Falso Negativo	Verdadeiro Negativo
Totais		P	N

Figura 3: Matriz de confusão gerada a partir de um conjunto de instâncias

Para entender melhor o preenchimento de uma matriz de confusão, considere que um determinado problema tem como principal objetivo classificar a predominância de uma determinada cor na imagem, e essa cor pode ser preta ou branca. A Figura 4 mostra as imagens que serão classificadas. Novamente tendo em mãos o classificador e a imagem que deseja classificar, deseja obter o resultado final da matriz de confusão. A primeira imagem a ser classificada (Figura 4 (a)), tem a predominância da cor branca, e foi classificada como branca. Então incrementa o valor de verdadeiros positivos, como é possível visualizar na Figura 5 (a).

A próxima imagem a ser classificada (Figura 4 (b)) também é uma imagem com predominância da cor branca e é classificada como preta. O resultado dessa classificação implica em um falso negativo como pode ser observado na Figura 5 (b). A terceira imagem a ser classificada Figura 4 (c) possui uma imagem com predominância da cor preta, e é classificada como preta, isso implica em um verdadeiro positivo, conforme é possível visualizar na Figura 5 (c). E por fim a ultima imagem a ser classificada (Figura 4 (d), onde a predominância de cor é preta e é classificada como branca, isso implica em um falso positivo. O resultado final dessa matriz de confusão pode ser visualizado na Figura 5 (d), onde a diagonal principal indica o total de acertos para essa classificação, que no caso seria 2.

Nas equações abaixo serão utilizadas alguns siglas para calcular os valores das métricas. A sigla VP indica o número de verdadeiros positivos e a sigla P indica o número total de positivos. A sigla FP represente o número de falsos positivos e a sigla N o número total de negativos. A sigla VN indica o número total de verdadeiros negativos.

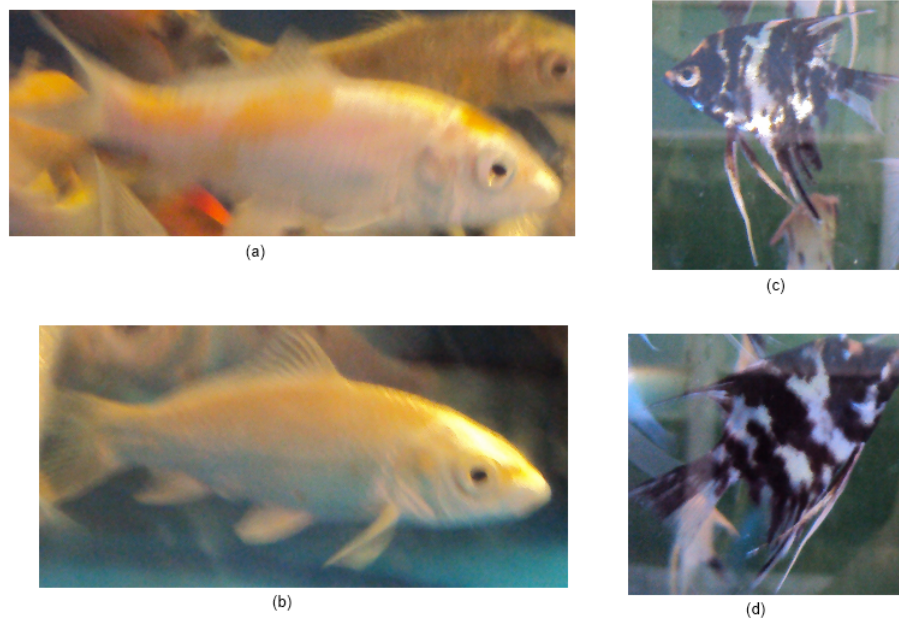


Figura 4: Matriz de confusão gerada a partir de um conjunto de instâncias

		Classe Verdadeira	
		branco	preto
Classificada Como	branco	1	0
	preto	0	0
Totais		Branco	Preto

(a)

		Classe Verdadeira	
		branco	preto
Classificada Como	branco	1	0
	preto	1	0
Totais		Branco	Preto

(b)

		Classe Verdadeira	
		branco	preto
Classificada Como	branco	1	0
	preto	1	1
Totais		Branco	Preto

(c)

		Classe Verdadeira	
		branco	preto
Classificada Como	branco	1	1
	preto	1	1
Totais		Branco	Preto

(d)

Figura 5: Matriz de confusão gerada a partir de um conjunto de instâncias

Taxa de verdadeiros positivos

A taxa de verdadeiros positivos consiste em verificar as amostras que são pertencentes a classe de verdadeiros e realmente foram classificadas como pertencentes a essa classe. Os resultados variam entre 0 e 1. O valor um representa que para todas as amostras classificadas que eram pertencentes a classe de verdadeiros foram classificados corretamente, o valor zero representa ao contrário. Como o número de instâncias positivas corretamente classificadas sempre será menor que - ou igual a - o número total de instâncias positivas e estamos dividindo o primeiro pelo segundo, nunca teremos valores fora do intervalo entre 0 e 1. A Equação 3.1 mostra como calcular esse valor. Considerado o exemplo anterior para realizar esse cálculo, o total de verdadeiros positivos foi 1, e total de amostras positivas

(brancas) é 2, resultando em 0,5.

$$taxaVerdadeirosPositivos = \frac{VP}{P} \quad (3.1)$$

Taxa de falsos positivos

A taxa de falsos positivos consiste em verificar qual o número de amostras pertencentes a classe de falsos e foram realmente classificadas como falsos. Esse valor também varia entre 0 e 1, o valor 1 represente que toda as amostras pertencentes a classe de falsos foram classificadas corretamente, o valor 0 representa ao contrário. A Equação 2 mostra como é calculado esse valor (SILVA, 2006). Utilizando o exemplo anterior para realizar os cálculos, é possível observar que o resultado final será 0,5, já que o total de falsos positivos foi 1 e o total de amostras negativas (pretas) é 2.

$$taxaFalsosPositivos = \frac{FP}{N} \quad (3.2)$$

Acurácia

Essa métrica é responsável por calcular a exatidão em que o problema se comportou baseado com o total de amostras. Os resultado pode variar entre 0 e 1, onde o valor um representa que o problema foi totalmente exato, e zero caso contrário. A Equação 3.3 mostra a forma como é calculada. Para o problema de classificação de imagem com predominância de cor, é possível observar que o valor de verdadeiros positivos e o valor de verdadeiros negativos ambos são iguais a 1, portanto a soma é 2. O total número de positivos (brancos) é 2, e o número total de negativos é 2, resultando na fração $\frac{(1+1)}{2+2}$ que é igual a 0,5.

$$acuracia = \frac{(VP + VN)}{P + N} \quad (3.3)$$

Precisão

A métrica precisão visa avaliar o quão foi preciso a classificação dos objetos. Assim como os demais acima citados, essa métrica também avalia entre 0 e 1. O valor 0 diz que a classificação não foi precisa, e o valor 1 indica que a classificação foi totalmente precisa. Uma informação muito importante que é levada em conta para essa métrica é quantidade

de erros obtidos. Essa informação acaba sendo umas das principais diferenças entre essa métrica e a métrica taxa de verdadeiros positivos. A Equação 5 mostra a fórmula de como calcular essa métrica. Para o exemplo anterior, é possível visualizar o resultado será 0,5 também, conforme a equação $\frac{(1)}{1+1}$.

$$precisao = \frac{VP}{VP + FP} \quad (3.4)$$

Medida-F

Essa métrica é a união dos resultados da taxa de verdadeiros positivos e precisão (QUINTA, 2013). O resultado final dessa métrica também varia entre 0 e 1. A Equação 3.5 exibe a forma de calcular essa métrica. Com os resultados das métricas anteriores é possível calcular o resultado dessa métrica para o exemplo acima citado. Com os resultados anteriores, a fração ficaria com os valores $\frac{2(0,5*0,5)}{(0,5*0,5)}$ resultado em 0,5.

$$medidaF = \frac{2(precisao * taxaVerdadeirosPositivos)}{precisao + taxaVerdadeirosPositivos} \quad (3.5)$$

Todas as métricas acima citadas são utilizadas apenas para problemas de duas classes. Devido a quantidade de classes utilizadas nesse trabalho (5 classes), as métricas acima tiveram que ser adaptadas. Para entender melhor, considere que agora tenhamos esse problema que está sendo estudado - classificação de espécies de peixes - e existam 5 espécies (classes) diferentes. Como é possível visualizar na Figura 6, a matriz passa a ser 5x5. Para essa situação, tendo em mãos o classificador e a imagem que deseja classificar resultará em 25 situações distintas. Se tivermos em mãos uma imagem do Peixe1, essa imagem pode ser classificada como Peixe1, Peixe2, Peixe3, Peixe4 e Peixe5. Caso a imagem seja do Peixe2, essa imagem pode ser classificada como Peixe1, Peixe2, Peixe3, Peixe4 e Peixe5. Se fizer isso para todas as classes, será possível visualizar as 25 situações distintas ao final da classificação. Essa possibilidade pode ser observada através da Figura 6.

A Figura 6 ilustra a matriz de confusão para o problema, já com os resultados obtidos. Nesta mesma figura é possível observar que cada linha da matriz representa o total de

		Classes atribuídas pelo classificador				
Classes reais		Peixe 1	Peixe 2	Peixe 3	Peixe 4	Peixe 5
	Peixe 1	15	0	0	4	0
	Peixe 2	2	12	0	10	0
	Peixe 3	3	0	8	0	0
	Peixe 4	7	3	1	26	0
	Peixe 5	0	6	0	3	0

Figura 6: Resultado da matriz de confusão para um problema contendo 5 classes

amostras de cada classe, ou seja, o valor de positivos. As métricas são calculadas sobre cada classe, por exemplo, o peixe1 teria o seu próprio valor de (FN_1, VP_1, P_1, N_1). Para a primeira linha da matriz (Peixe1), se realizar o somatório de todos os valores resultará em 19. Esse valor representa o número de total de amostras daquela classe. Para verificar o número de verdadeiros positivos (VP_1) basta observar a célula que há um “casamento” entre a classe real e a classe atribuída pelo classificador. O valor de N_1 é calculado realizando uma subtração entre o total de amostras do conjunto e as amostras positivas. Para calcular o valor de falsos negativos, basta realizar uma subtração entre o total de positivos daquela classe (P_1) e o verdadeiros positivos (VP_1). Para calcular o valor de verdadeiros negativos, basta realizar uma subtração entre o valor de negativos (N_1) e o valor de falsos positivos (FP_1) referentes aquela classe. Uma amostra é classificada como falso positivo quando não é pertencente a determinada classe, por exemplo, o número de falsos positivos da quarta linha (Peixe4) da Figura 6 seria o valor 11. Ao final da classificação é realizada uma média entre os valores de cada variável. Por exemplo, para a métrica taxa de verdadeiros positivos, é calculado a média entre os valores de todas as classes (VP_1, VP_2, VP_3, VP_4, VP_5), obtendo assim um sexto resultado que seria a média (VP_M). A equação 3.6 mostra a fórmula para calcular a medida-F.

$$medidaF = \frac{2(precisao * VP_M)}{(precisao + VP_M)} \quad (3.6)$$

Para esse problema foi considerado um total 100 imagens. Ao final da classificação foi possível observar que 61 imagens foram classificadas corretamente. É possível observar esse valor na diagonal principal onde o classificador atribuiu o valor correto da classe real,

assim sendo considerado como um acerto.

3.4 Teste de Friedman

O teste de Friedman é um teste não paramétrico e é executado para analisar os resultados de um determinado delineamento experimental. O teste de Friedman é usado quando há mais de duas condições de emparelhamento e deseja-se saber qual obteve um melhor resultado. O teste de Friedman não usa os dados diretamente, mas sim a ordem que cada um ocupa depois de serem ordenados de forma crescente. Toda a ordenação é feita de forma separada para cada uma das combinação e não para o conjunto todo.

Um valor, chamado valor-p é calculado e dependendo do resultado desse valor a hipótese pode ser rejeitada ou não. Com o valor-p é possível aceitar ou rejeitar uma hipótese de acordo com o grau de significância que o teste irá adotar. Por exemplo, se para esse problema - reconhecimento de espécies de peixes - determinamos que o valor de significância será 10% e o valor obtido por alguma métrica for 5%, é possível afirmar que a hipótese nula pode ser rejeitada, devido ao valor de p ser inferior ao valor de significância. Rejeitando a hipótese nula, pode-se dizer que as comparações são diferentes, porém não há nenhuma informação sobre qual é melhor que qual. Realizando uma análise pós teste, é possível ver qual das combinações é melhor (QUINTA, 2013).

3.5 Análise pós-teste

Após a realização o teste de Friedman um valor-p é obtido, o que significa que alguns dos grupos de dados da entrada tem distribuições diferentes uns dos outros, porém não é possível saber qual deles se diferem. A análise pós-teste (ou *post hoc*) tem como principal objetivo “descobrir” qual(is) o(s) par(es) que tem uma diferença estatística.

A análise pós-teste consiste em verificar qual par de combinações se diferem estatisticamente. A comparação é feita de 2 em 2 ($n > 2$). Por exemplo se há 4 combinações, serão realizadas 6 comparações (1-2, 1-3, 1-4, 2-3, 2-4, 3-4) para encontrar em qual das comparações há uma diferença estatística.

4 *Principais espécies ornamentais da cidade de Campo Grande - MS*

Com a grande variação de espécies de peixes encontradas nos aquários ornamentais e nos rios de todo o Brasil, a classificação de diferentes espécies de peixes por um equipamento independente (dispositivo móvel) é sempre um desafio a ser superado. Segundo o Instituto Chico Mendes de Conservação da Biodiversidade (ICMBio)¹, o Brasil possui cerca de 5164 espécies de peixes distribuídos nos diversos tipos de biomas existentes. Ainda segundo o Instituto Chico Mendes de Conservação da Biodiversidade (ICMBio), o estado do Mato Grosso do Sul está dividido entre os biomas do pantanal e do cerrado totalizando cerca de 1500 espécies de peixes.

Devido a dificuldade de obter diversas imagens de espécies nativas do estado, este trabalho tratará inicialmente somente com espécies ornamentais, devido a captura de imagens de tal espécie ser mais simples e com isso já ser possível obter um resultado sobre o trabalho. Após uma pesquisa com dois proprietários de lojas especializadas na venda de peixes ornamentais, foi concluído que as 7 espécies mais comuns são as seguintes:

- **Beta** (*Beta Splends*): Conhecido também como peixe briga, este tipo de peixe é muito agressivo com machos da mesma espécie, porém com peixes de outra espécie o comportamento é um pouco mais pacífico. Tanto o macho quanto a fêmea dessa espécie pode chegar a 10 cm de comprimento. O macho e a fêmea são diferenciados pelo tamanho da nadadeira caudal, sendo a do macho mais extensa. O Beta deve ser criado em aquários não muito altos, pois o mesmo sobe até a superfície para respirar. A temperatura média deve ser por volta de 27 graus Celsius e com um pH de 7,0. Devido a pouca quantidade de peixes disponíveis no ambiente de captura essa espécie não está presente no banco de imagens.

¹<http://www.icmbio.gov.br/portal/>



Figura 7: Espécie de peixe Beta.

- **Platy** (*Xiphophorus maculatus*): Uma espécie muito comum no meio dos aquariófilas² por ser espécies não muito exigente e bastante sociável. A sua rusticidade de cores é uma das formas pela qual chama atenção dos aquariófilas. Sua origem foi na América central, mais precisamente no México, Guatemala e Honduras³. A água deve ser mantida entre 22 e 27 graus celsius, com um pH de aproximadamente 7.5. Esta espécie de peixe gosta de águas calmas e de preferência com plantas naturais. Reproduzem-se facilmente em aquários, e quando adultos podem atingir até 6cm.



Figura 8: Espécie de peixe Platy.

- **Kinguio** (*Carassius auratus*): Coloridos e brincalhões os peixes da espécie *Carassius auratus* são muito populares em todo o mundo. São peixes bastantes sociáveis, e é possível mantê-los em um aquário com outras espécies pois dificilmente haverá algum

²Pessoas que cuidam e estudam sobre as mais variadas espécies de peixes.

³<http://www.wazlawik.com.br/index.php/peixes.html>

problema. Os kinguios não são peixes exigentes, porém o aquário deve ser mantido sempre limpo para um melhor desenvolvimento deste tipo de peixe. Hoje em dia, existem mais de 10 variações dessa espécie⁴, em diferentes tamanhos e colorações. Seu tamanho varia de acordo com o local onde é criado, se criado em um aquário seu tamanho médio é de 15cm, já quando criado em lagos seu tamanho pode chegar até 30cm.



Figura 9: Espécie de peixe Kingiuo.

- **Guppy** (*Poecilia reticulata*): Uma espécie muito resistente e indicado para aquarífilistas iniciantes. Uma das características mais chamativas desse tipo de espécie é o tamanho da cauda em relação ao corpo, e os mais diferentes tipos de colorações existentes. As barbatanas das fêmeas não são tão grandes e tão atraentes (coloridas) se comparadas com as dos machos. O macho, em média, chega a 4cm de comprimento, já a fêmea pode chegar até 7cm. A temperatura da água para essa espécie deve ser mantida em torno de 27 graus celsius e com um pH de 7.3. Devido a pouca quantidade de peixes disponíveis no ambiente de captura essa espécie não está presente no banco de imagens.
- **Tricogaster azul** (*Trichogaster trichopterus*): Tricogaster azul é uma espécie muito resistente, porém deve ter alguns cuidados na criação desse peixe. O primeiro ponto importante sobre essa espécie é que pode chegar até 15cm de comprimento, assim tendo que ter cuidado com o tamanho do aquário já que podem superpopularizar o

⁴<http://falandodepeixes-kingiuo.blogspot.com.br/>

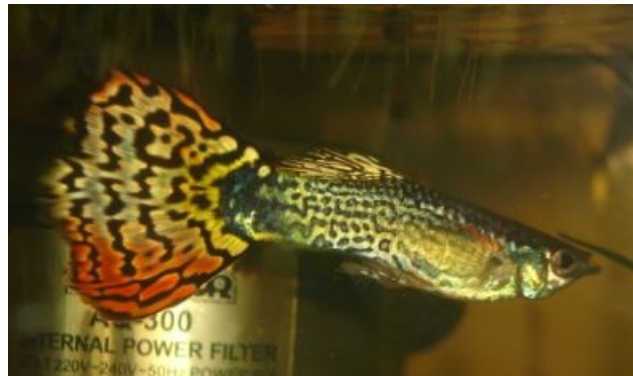


Figura 10: Espécie de peixe Guppy.

aquário. Outro ponto importante é que os machos tendem a ser muito territoriais e se tiver com outras espécies menores no mesmo aquário podem atormentá-las. A identificação dos sexos é muito simples, o macho tem uma barbatana dorsal comprida e pontuada, já a fêmea possui uma barbatana curta e arredondada. Essa espécie deve ser mantida em uma média de temperatura de 27 graus celsius e a água com um pH de 6.8.



Figura 11: Espécie de peixe Tricogaster azul.

- **Acará Bandeira** (*Pterophyllum scalare*): Acará bandeira é uma espécie considerada pelos aquaríofistas não muito exigente, podendo ser criada por aquaríofistas iniciantes sem muitos problemas. Essa espécie possui várias colorações diferentes, entre elas está a que é totalmente preta, parcialmente preta, de coloração ouro entre outras⁵. Todas as variações pertencem à mesma espécie. Para esse trabalho foi utilizada a variação conhecida como palhaço. Para essa espécie é recomendando manter a temperatura da água em 20 graus Celsius. Como é possível visualizar na Figura 12, esse tipo de peixe possui o corpo bem achatado lateralmente, podendo chegar a 15 cm de comprimento, sendo que geralmente os machos são maiores que as fêmeas.

⁵<http://www.pisciculturacristal.com.br/acara-bandeira.html>



Figura 12: Espécie de peixe Acará Bandeira.

- **Molinésia Preta** (*Poecilia sphenops*): Molinésia Preta, também conhecida como Molinésia Negra e Molly Preta, é um peixe extremamente ativo, e muitas vezes acabam sendo agressivos com peixes da mesma espécie. Também é um peixe comum no meio dos aquariófilistas devido a sua facilidade de ser criado e sua pouca exigência em relação a espaço. Essa espécie tem uma boa convivência com peixes de outras espécies. A temperatura recomendada para se manter um peixe dessa espécie é entre 24 graus Celsius até 28 graus Celsius. É recomendando também que o pH da água para esse espécie esteja entre 7.2 e 7.8. A Figura 13 mostra uma peixe da espécie descrita.



Figura 13: Espécie de peixe Tricogaster azul.

5 *Técnicas de Identificação de pontos de interesse*

Para compreender melhor o conceito de pontos de interesse, basta olhar para a Figura 14 e observar os círculos que são justamente os pontos de interesse. Resumidamente, são pontos que melhor descrevem uma determinada imagem.

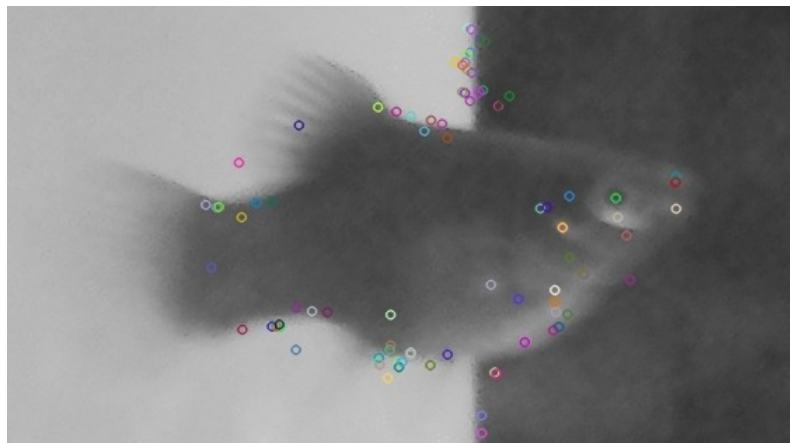


Figura 14: Peixe da espécie platy sangue. Os círculos existentes em várias áreas da figura, mostram os pontos de interesse.

Os dois algoritmos que serão descritos abaixo são bons para realizar esse trabalho com um desempenho satisfatório. O motivo da escolha desses algoritmos é pelo seu desempenho satisfatório, uma vez que para obter sistemas de Visão Computacional robustos, essa medida é importante. Primeiramente será explicado o algoritmo SIFT e posteriormente será explicado o algoritmo SURF. Alguns conceitos importantes do algoritmo SIFT estarão na seção do algoritmo SURF.

5.1 SURF: Speeded-Up Robust Features

Inspirado no SIFT, o algoritmo SURF é um sólido detector de pontos de interesse em imagens, porém com um custo computacional menor (RACHID; PEREIRA, 2009). Diferentemente dos algoritmos apresentados anteriormente, que a partir de uma imagem gerava um vetor de atributos, o algoritmo SURF, assim como o SIFT, trabalha um pouco diferente, com dois principais elementos:

- Detecção de pontos de interesse;
- Descrição dos pontos de interesse.

O algoritmo SURF utiliza alguns conceitos matemáticos importantes, entre eles está o de derivadas. Este conceito é essencial, pois através dele é possível encontrar variações em uma determinada imagem, utilizando derivadas parciais por exemplo.

Esse algoritmo consiste em basicamente detectar os pontos de interesse utilizando imagens integrais e matriz Hessiana e depois utiliza o conceito de supressão de não máximo em 3D para detectar os pontos de interesse em diversas escalas. Após a detecção, inicia o processo de descrição, utilizando-se os conceitos de cálculo de orientação dominante para invariância a rotação, distribuição de respostas de Haar Wavelets depois gera um vetor de 64 atributos para cada ponto de interesse. E por fim é realizado a classificação da imagem, utilizando o conceito de casamento de modelos.

O conceito importante que aparece no algoritmo é o de Funções *Gaussianas*. Através das Funções *Gaussianas* quando se analisa um determinado ponto de uma imagem é possível dar mais importância aos *pixels* que estão mais próximo daquele ponto, sem desconsiderar os outros *pixels*, mas sim diminuindo o peso daqueles que estão mais distantes do ponto.

5.1.1 Detecção de pontos de interesse

A detecção de pontos interesses, segundo Bay et al. (Junho 2008), consiste em encontrar pontos relevantes para comparar com uma outra imagem, para saber por exemplo, se as imagens são iguais ou diferentes, ou se ela é mais próxima de uma imagem, ou até mesmo encontrar ocorrências de uma determinada forma.

5.1.1.1 Imagens Integrais

Este conceito foi utilizado no algoritmo SURF como um meio rápido e eficaz para calcular a soma dos valores (valores em *pixel*) de uma imagem. Este método consiste em gerar a partir da imagem original outra imagem onde cada ponto será a soma de todos os pontos que estão acima e a esquerda dele até o ponto O (Figura 15). Este cálculo é realizado apenas uma vez.

Imagine agora que é necessário saber a área de uma parte da imagem, conforme a Figura 15, isto se torna uma tarefa simples, pois basta realizar apenas três somas e quatro acessos a memória.

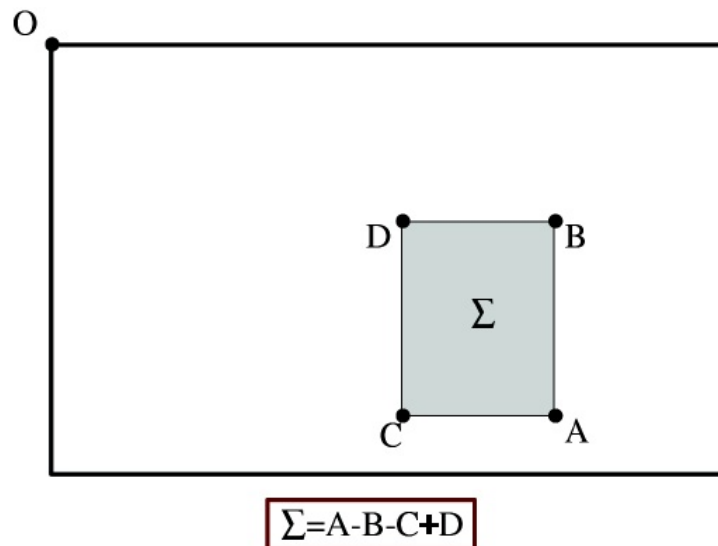


Figura 15: Utilizando imagens integrais, são necessárias apenas três adições e quatro acessos à memória para calcular a soma de intensidades dentro de uma região retangular de qualquer tamanho.

Conforme é possível observar na Figura 15, o cálculo independe do tamanho da imagem pois de qualquer maneira serão realizados apenas quatro cálculos. Isto é importante na abordagem do algoritmo pois muitas vezes utiliza-se imagens grandes (BAY et al., Junho 2008).

5.1.1.2 Matriz Hessiana baseada nos pontos de interesse

O algoritmo é baseado na matriz Hessiana devido ao seu bom desempenho e precisão. A matriz Hessiana é formada pelas derivadas parciais nas direções vertical, horizontal

e diagonal, conforme é possível observar na Figura 16. Uma das possíveis formas de saber se está ocorrendo variação em uma determinada região da imagem, é calculando a matriz Hessiana, que nada mais é do que calcular as derivadas parciais da função. Com o cálculo do determinante da matriz, é possível saber se está ocorrendo variação ou não em uma determinada região, ou seja, se o determinante for um valor baixo, a variação numa determinada região é baixa, caso seja um valor alto, a variação é alta.

$$\mathcal{H}(\mathbf{x}, \sigma) = \begin{bmatrix} L_{xx}(\mathbf{x}, \sigma) & L_{xy}(\mathbf{x}, \sigma) \\ L_{xy}(\mathbf{x}, \sigma) & L_{yy}(\mathbf{x}, \sigma) \end{bmatrix}$$

Figura 16: Matriz Hessiana. $\mathbf{x}$ representa um ponto da imagem e σ determina a escala. L_{xx} derivada parcial no eixo x , L_{yy} derivada parcial no eixo y e L_{xy} derivada parcial no eixo xy .

Conforme mostra Bay et al. (Junho 2008), o algoritmo SURF é um algoritmo que busca eficiência, fazendo uma busca por pontos de interesses que são realmente relevantes. Segundo Bay et al. (Junho 2008), para realizar a aproximação da Hessiana, o algoritmo SIFT realiza a aproximação utilizando a Laplaciana da Gaussiana, conhecida também como chapéu mexicano, quando realiza-se a convolução desta função com a função original são obtidos valores altos onde se tem uma grande variação. Para aplicar a Laplaciana da Gaussiana em uma imagem através de convolução, é necessário definir o núcleo de convolução, ou seja, definir uma matriz que é usada para realizar a convolução. Após varrer toda a imagem, *pixel* à *pixel*, será dada uma resposta da convolução que será mais alta naquelas regiões em que há maior variação.

Conforme Bay et al. (Junho 2008), o algoritmo SURF vai mais além para realizar a aproximação, ou seja, é menos rigoroso na hora de encontrar a Hessiana (Figura 17). O algoritmo realiza uma aproximação um pouco mais grosseira utilizando *Box Filter* ou Filtros Caixa, conforme é possível observar na Figura 17.

A Figura 18 mostra a fórmula aproximada para calcular o determinante da matriz Hessiana (Figura 16). Utilizando convolução é possível obter uma aproximação do determinante da matriz Hessiana. Sobre cada ponto da imagem original é calculado o determinante, e o resultado indica o quanto de variação está ocorrendo naquele ponto de imagem.

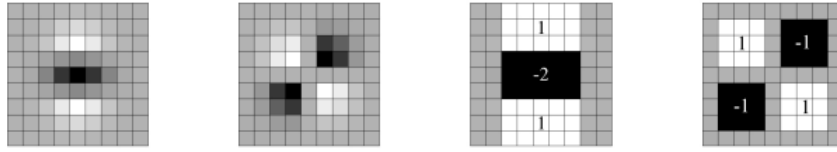


Figura 17: Da esquerda para a direita: Derivadas parciais de segunda ordem na direção x e xy e suas aproximações, utilizando filtros de caixa. As regiões em cinza são iguais a zero.

$$\det(\mathcal{H}_{\text{approx}}) = D_{xx}D_{yy} - (wD_{xy})^2.$$

Figura 18: Fórmula utilizada para realizar a aproximação do determinante. D_{xx} derivada parcial no eixo x, D_{yy} derivada parcial no eixo y e D_{xy} derivada parcial no eixo xy.

5.1.1.3 Pirâmide de escala e oitavas

Bay et al. (Junho 2008) dizem que pontos de interesse precisam ser encontrados em qualquer escala, independentemente do tamanho. Não basta ter pontos de interesse em somente uma escala, para ter um reconhecimento preciso, é necessário encontrar pontos de interesse em várias escalas.

Uma das possíveis soluções é, a partir de uma original, gerar várias outras imagens em escalas diferentes, conforme mostra a Figura 19, utilizando operações de *zoom*. Quando necessita-se buscar um ponto de interesse, em vez de realizar a busca apenas na imagem original, é necessário realizar a busca em toda a pirâmide, conseqüentemente buscando em todas as escalas disponíveis.

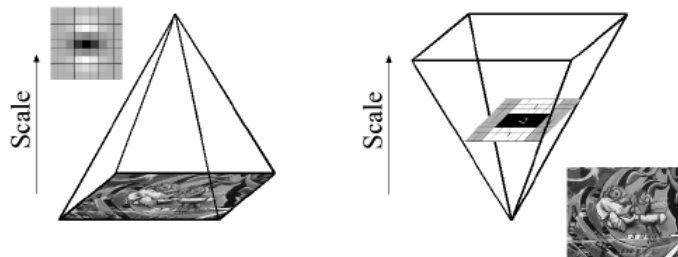


Figura 19: Em vez de utilizar a redução do tamanho da imagem (esquerda), a utilização de imagens integrais permite o aumento de escala do filtro por um custo constante (direita).

De acordo com Bay et al. (Junho 2008), o algoritmo SURF realiza a busca aumentando o tamanho do filtro da imagem (Figura 20). Para obter tal resultado, realiza-se a convolução aplicando diferentes filtros. Uma maneira de fazer isso é diminuir pela metade o tamanho, tanto em x como em y, até chegar em um ponto onde não é possível mais reduzir, ou não faz sentido ter uma imagem de tal tamanho.

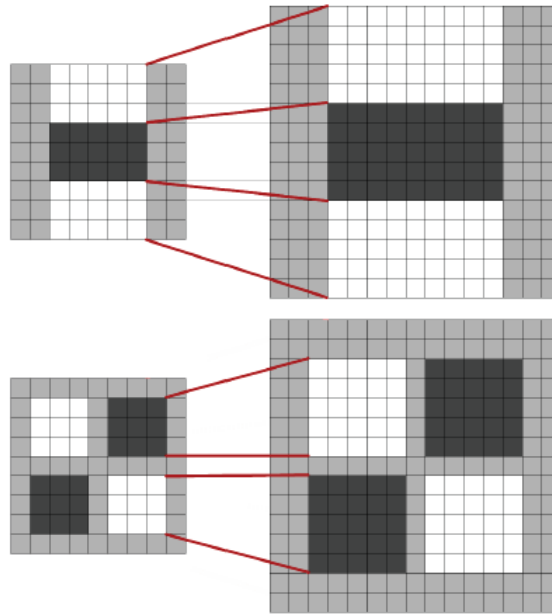


Figura 20: Exemplo de filtros em escalas diferentes, 9x9 (esquerda) e 15x15 (direita)

Ainda conforme Bay et al. (Junho 2008) o crescimento do tamanho é exponencial e neste momento aparece o conceito de oitavas. O conceito de oitavas consiste em agregar valores intermediários entre uma escala e outra (Figura 21) e não apenas dobrar o tamanho (escala). Observe que na Figura 21 existem valores ímpares, para os quais são possíveis pegar o centro da imagem. Considere o valor 9, neste caso o irá ficar 4 para cada lado, ficando um valor vago no meio e com isso é possível pegar o centro da imagem. Também na Figura 21 é possível observar que para escala 1, tem-se os valores 9, 15, 21 e 27, já para a escala 2 tem os valores 15, 27, 39, 51 e assim por diante.

5.1.1.4 Supressão de não máximos em 3D

Este é o último passo para detectar os pontos de interesse. Supressão de não máximos em 3D consiste em encontrar pontos que tenham valores altos em relação aos vizinhos, ou

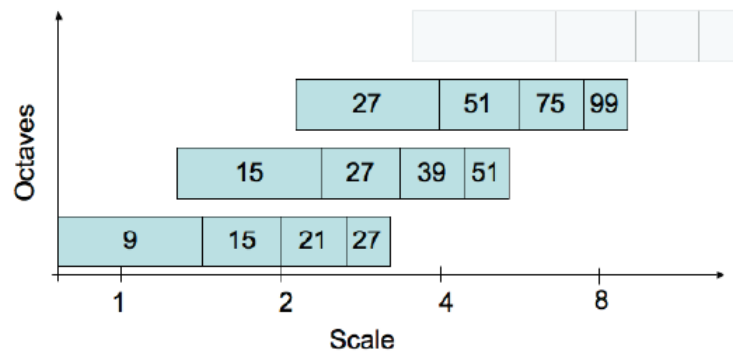


Figura 21: Representação da escala de oitavas. O eixo horizontal representa o logaritmico das escalas. O eixo vertical representa as oitavas.

seja, os pontos que sobressaem em relação a sua vizinhança. É realizado tanto na escala superior quanto na inferior, quando tiver, daí que se origina-se o 3D. Uma maneira de saber se há escalas superiores e inferiores disponíveis é olhando para a escala de oitavas (Figura 21). Depois de aplicar a supressão de não máximos em 3D, serão desconsiderados os pontos que estão ao lado do ponto máximo, ou seja, serão suprimidos os outros valores.

A Figura 22 representa um exemplo de resultado encontrado após a execução da primeira parte do algoritmo. Na figura citada é possível observar que há alguns círculos, o centro desses círculos representam um ponto interessante que foi encontrado. O tamanho do círculo está diretamente ligado a escala em que o ponto foi encontrado.

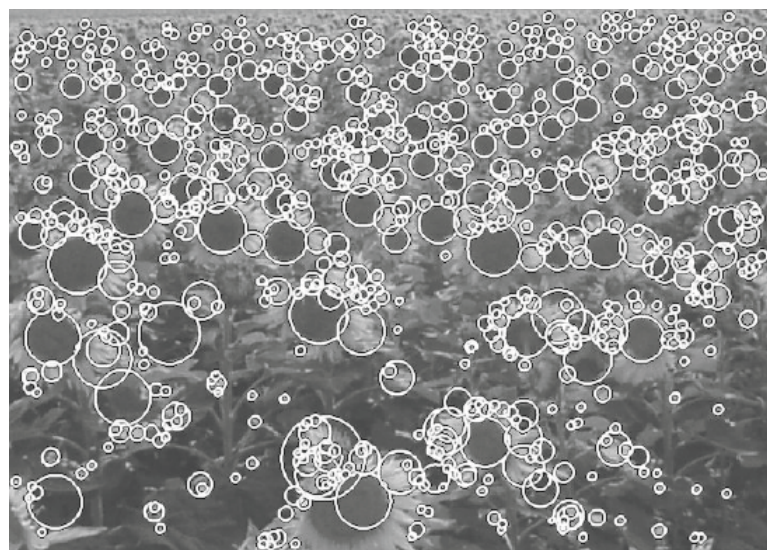


Figura 22: Exemplo real de detecção de pontos de interesse

5.1.2 Descrição dos pontos de interesse

Segundo Bay et al. (Junho 2008) a descrição dos pontos de interesse é uma fase importante tal qual a detecção dos pontos de interesse. Após realizar a detecção dos pontos, é preciso descrever esses pontos.

5.1.2.1 Cálculo de orientação dominante para invariância à rotação

Ainda segundo Bay et al. (Junho 2008), para cada ponto de interesse (Figura 23) encontrado cujo tamanho está relacionado com a escala, calcula-se a resposta da convolução para os dois filtros (Figura 24). Com isso é possível observar as variações que estão ocorrendo na direção x e na direção y (Figura 25).



Figura 23: Representação do descritor de pontos de interesse. Note que os pontos interessantes estão marcados com um quadrado e a linha reta que vai do centro a algum lado é a direção onde ocorre maior variação.



Figura 24: Filtros em x (esquerda) e filtros em y (direita)

Como é possível observar na Figura 25 há pontos com uma grande variação em x , porém em y é nula e vice-versa. Também existem pontos em que ocorre variação tanto em x como em y . Conforme o gráfico representado na Figura 25, onde a parte da pizza tiver o valor maior (tamanho dos pedaços fixo) indica a ocorrência de uma maior variação. Com a técnica acima explicada, é possível garantir que se rotacionar uma imagem por exemplo, ainda é possível encontrar padrões nela.

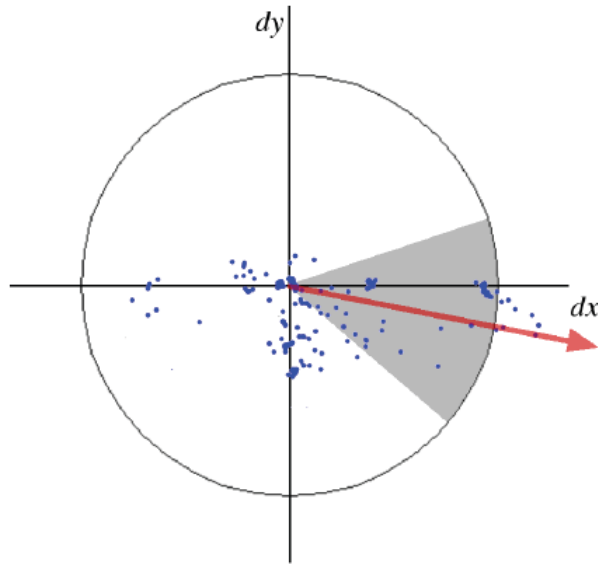


Figura 25: Gráfico em formato de pizza que representa a resposta, após aplicar os filtros. Tamanho da pizza é fixo.

5.1.2.2 Distribuição de respostas de Haar Wavelets

Esta etapa consiste em pegar o ponto de interesse identificado e a partir do centro, calcular ou imaginar um quadrado conforme a Figura 26. Tal quadrado estará alinhado com a direção que foi calculada no passo anterior. Esse quadrado será dividido em subquadrados menores, gerando 16 quadrados menores. Para cada quadrado menor calculado, calcula-se novamente a resposta da convolução, utilizando novamente os núcleos acima citado (Figura 24). Lembrando que os quadrados referentes aos núcleos de convolução, agora ficaram alinhados com os 64 quadrados menores, assim o conceito horizontal e vertical estará ligado com a direção principal.

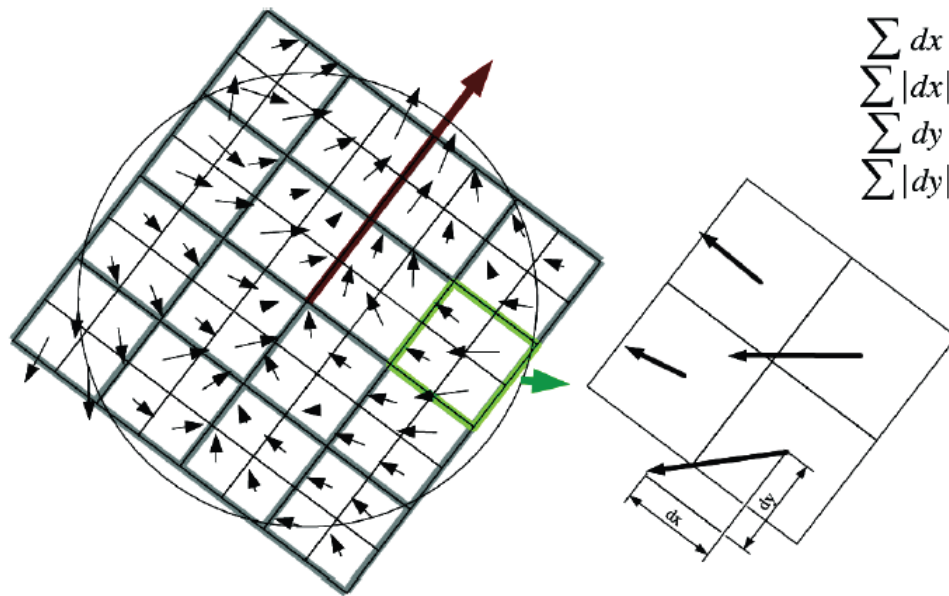


Figura 26: Representação do quadrado de respostas de Wavelet

Quando se utiliza os núcleos está sendo feita a aproximação de um conceito um pouco mais complexo que é as *Haar Wavelets*. *Haar* é um conjunto de funções. Porém, é possível obter o mesmo resultado utilizando convolução.

5.1.2.3 Vetor de 64 atributos para cada ponto de interesse

Os somatórios representados na Figura 26 indicam o resultado em tais direções para cada quadrado 2x2. No final deste procedimento tem-se 16 quadrados, gerando assim um vetor 64 posições ($16 \times 4 = 64$). Com isso para cada ponto de interesse encontrado, terá um vetor de 64 posições descrevendo este ponto (Figura 27). Pelo fato do algoritmo de SURF gerar no final um vetor com 64 posições para cada ponto de interesse, ele é considerado um descritor de pontos de interesse baseado em um vetor 64 posições. Também pelo motivo de gerar um vetor de 64 posições para cada ponto de interesse, que não é possível aplicar algoritmos clássicos de aprendizagem supervisionada depois do SURF.

5.2 SIFT: Scale-invariant Feature Detector

SIFT é um algoritmo para reconhecimento de padrões publicado no ano de 1999 e proposto por Lowe (2004). Conforme Lowe (2004), para gerar o conjunto de características

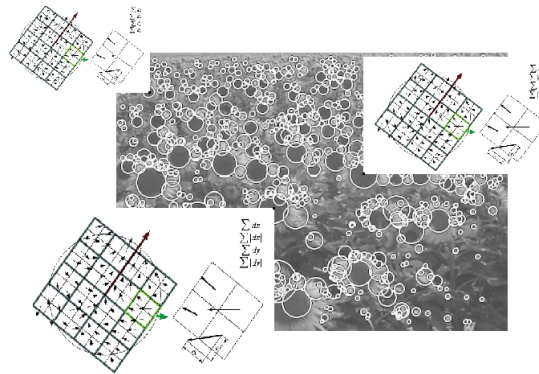


Figura 27: Exemplo geração de vetor de atributos para cada ponto de interesse

serão necessários os quatros principais cálculos a seguir: a detecção do espaço de escala, localização dos pontos de interesse, atribuição da orientação e descrição dos pontos chaves.

Segundo Lowe (2004), as características extraídas a partir desse algoritmo são invariantes a escala e a rotação, com isso é possível fornecer uma correspondência um tanto robusta. Além disso o algoritmo é capaz de fornecer informações importantes sobre a mudança na posição da imagem, da iluminação e da quantidade de ruídos presentes nessa imagem. Ainda segundo Lowe (2004), as características fornecidas ao final da execução desse método são altamente distintas uma das outras.

Para realizar a detecção dos espaços da escala e os pontos de interesse invariantes a escala e a rotação, são utilizadas as funções Gaussianas. A localização dos pontos de interesse é feita utilizando uma expansão de Taylor da função diferença de Gaussianas aplicada a imagem (LOWE, 2004). Após a localização dos pontos de interesse é atribuída a orientação para esses pontos, obtida através das características locais da imagem. Por fim, é realizado a construção do descritor, na qual os pontos de interesse são transformados em uma representação, onde é possível atribuir níveis mais significativos conforme a distorção e as alterações de iluminação (LOWE, 2004).

5.2.1 Detecção do espaço de escala

A primeira etapa do algoritmo consiste em buscar pontos que sejam invariantes a alteração da escala, de tal forma que independentemente da posição da câmera esses pontos sejam detectados. Para isso, o algoritmo procura na imagem características que sejam estáveis em diferentes escalas (LOWE, 2004).

Para encontrar esses pontos em diferentes escalas, é utilizada uma função conhecida como espaço de escala, que é uma função Gaussiana. Segundo Lowe (2004), a função de espaço de escala é definido por $L(x, y, \sigma)$ que é produzida através da convolução de uma variável de escala Gaussiana $G(x, y, \sigma)$ e a imagem de entrada $I(x, y)$, conforme é mostrado na Equação 5.1, onde $G(x, y, \sigma)$ é exibido na Equação 5.2. A Figura 28 exibe um exemplo de imagem aplicando esses filtros.

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad (5.1)$$

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} * e^{-(x^2+y^2/2\sigma^2)} \quad (5.2)$$

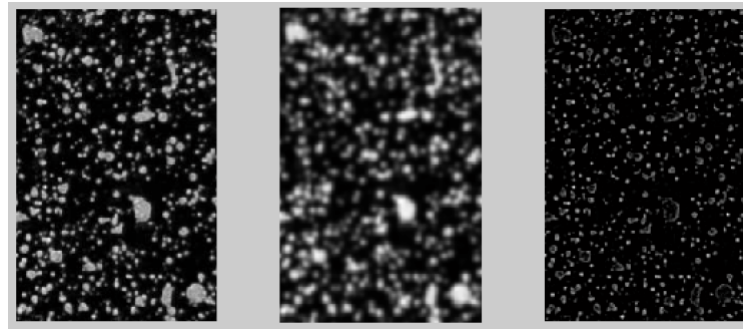


Figura 28: Exemplo de aplicação do filtro para a detecção do espaço de escala. Esquerda imagem original. Centro filtro Gaussiano. Direta imagem resultante do filtro da diferença de Gaussianos.

A fim de aumentar a eficiência pela busca de pontos de interesse é utilizada uma função denominada DoG (Difference of Gaussian) que é calculada a partir de duas escalas próximas (LOWE, 2004). Segundo Lowe (2004), entre as diversas qualidades desta função está que é extremamente eficaz. A Figura 29 ilustra como é feito esse cálculo.

Após a realização de todos os passos da função DoG, é feita a detecção dos extremos para cada intervalo de oitava. Os extremos são obtidos comparando a intensidade de cada ponto, com os seis oitos vizinhos para a escala que está sendo observada.

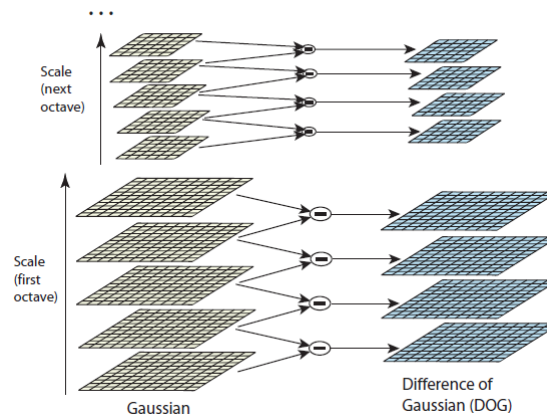


Figura 29: Representação utilizada por Lowe (2004) para representar como é obtido as DoG para diversas oitavas de uma determinada imagem.

5.2.2 Localização dos pontos de interesse

Todos os pontos detectados anteriormente são fortes candidatos a serem pontos de interesse. Com todos os pontos detectados anteriormente, deseja-se calcular a localização exata para cada um destes pontos. Utilizando expansão de Taylor da função DoG aplicada a imagem, é possível determinar a localização interpolada do máximo (LOWE, 2004).

Ainda segundo Lowe (2004), os pontos que foram detectados anteriormente e possuem baixo contraste podem ser rejeitados, porque são sensíveis a ruídos ou estão mal localizados ao longo de uma borda. Para chegar a essa conclusão sobre um ponto, são realizados diversos cálculos e esses cálculos são mostrados por Lowe (2004). A Figura 30 mostra as imagens resultantes do processo.

5.2.3 Atribuição da orientação

Visando a etapa de descrição dos pontos de interesse, essa etapa é responsável por atribuir uma orientação para cada ponto de interesse para construir pontos de interesse invariantes a rotação. Para obter essas características são utilizadas informações locais da imagem (LOWE, 2004).

Utilizando cada amostragem da imagem, calcula-se a magnitude e orientação do gradiente utilizando a diferença entre os *pixel*. As Equações 5.3 e 5.4 mostram como calcular a magnitude e orientação, respectivamente.

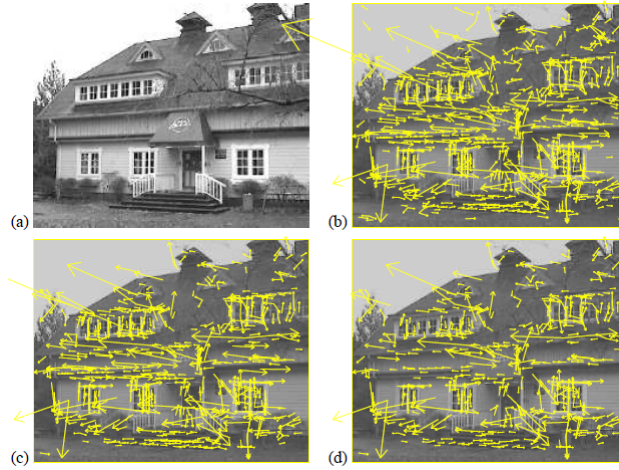


Figura 30: Mostra as etapas da seleção dos pontos de interesse. (a) imagem original 233x189. (b) os 832 pontos de interesse locais em máximos e mínimos da função DoG. (c) após a aplicação do limiar de contraste, restam 729 pontos de interesse. (d) após a retirada dos pontos que não interessam (possuem baixo contraste ou são sensíveis a ruídos) restam 536 pontos de interesse. Cada ponto de interesse é representado por uma flexa, que indica a direção, sentido e escala.

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2} \quad (5.3)$$

$$\theta(x, y) = \tan^{-1}(((L(x, y+1) - L(x, y-1)) / (L(x+1, y) - L(x-1, y)))) \quad (5.4)$$

Um histograma de orientação é construído a partir das orientações gradiente de pontos de amostragem dentro de uma região em torno de um ponto chave (LOWE, 2004). Os histograma possui 36 regiões, onde é possível cobrir todas as orientações possíveis (Figura 31). Os picos existentes na Figura 31 correspondem a orientação e magnitude de maior dominância. Além dos pontos de máximo, os pontos que possuem 80% do máximo podem ser utilizados para atribuir mais de uma orientação ao mesmo ponto.

5.2.4 Descrição dos pontos chaves

Com as etapas anteriores, foi possível construir um ponto de interesse invariante a escala e rotação. O próximo passo do algoritmo é calcular um descritor que seja extremamente distintivo, para a região da imagem. Com informações da magnitude e orientação

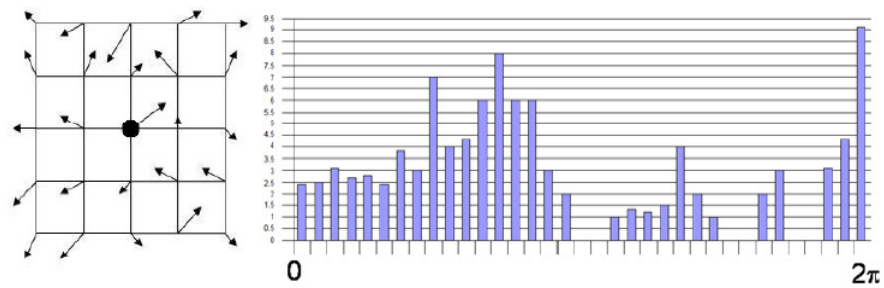


Figura 31: Histograma de orientações para cada pontos de interesses.

dos gradientes é criado então o descritor do ponto de interesse. A Figura 32 mostra o descritor já criado, onde as setas representam informações de escala e orientação.

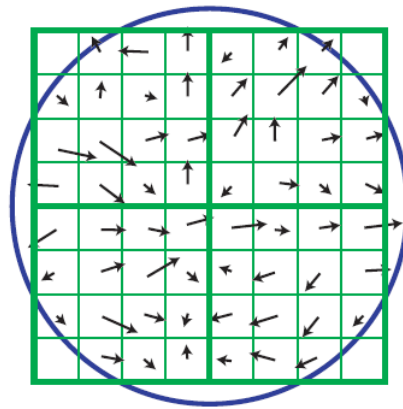


Figura 32: Exemplo de um descritor para um ponto de interesse.

A Figura 33 mostra o resultado do processo. Como é possível observar utilizando toda a imagem utilizando as informações da imagem a esquerda, calcula-se qual é a orientação e escala predominante para aquele gradiente. Esse cálculo é feito utilizando funções Gaussianas (LOWE, 2004). Ao final do processo, um descritor 2x2 é obtido, já com as informações predominante em relação a escala e orientação.

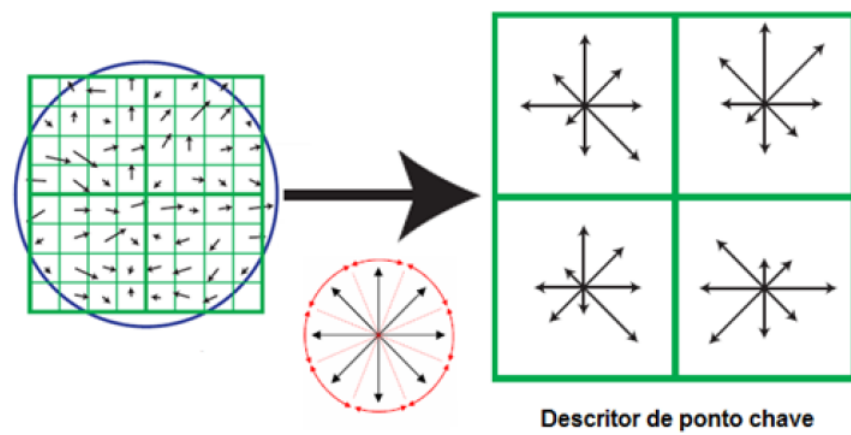


Figura 33: Construção de pontos de interesse proposto por Lowe (2004). Descritor 2x2 utilizando 48 elementos.

6 *Casamento de Modelos*

Após a realização de todas as etapas dos algoritmos citados acima, chega-se na etapa da classificação de uma determinada imagem. Para realizar esta etapa é necessário possuir um banco de imagens, para comparar com a imagem que deseja classificar. Para cada imagem desse banco de imagens, são extraídos os pontos de interesse (cada imagem pode possuir vários pontos de interesse) e um descritor para cada ponto com um vetor de 64 posições. Essa primeira fase é denominada de fase de treinamento, e nela são extraídos os pontos de interesse utilizando o algoritmo SURF ou SIFT e posteriormente os descritores. Com os descritores de cada imagem do conjunto de treinamento em mão, classifica-se uma determinada imagem. Para realizar a classificação, aplica-se um dos algoritmos estudados acima nesta imagem também, extraíndo os descritores e para cada imagem do conjunto de treinamento compara-se esta imagem.

Para entender melhor o funcionamento, observe na Figura 34 que tem peixes com os pontos de interesses detectados e outros não. Os pontos de interesses são detectados e descritos na primeira etapa, onde se aplica algum dos algoritmos para detecção dos pontos de interesse no conjunto de treinamento. Cada ponto de interesse detectado é um vetor de 64 posições. Após a fase de detecção e descrição dos pontos de interesse, imagine que chegue uma imagem (Figura 34 (i)), e deseja classificar-la. Para essa imagem, aplica-se o algoritmo para detecção de pontos de interesse com o objetivo de extrair os pontos de interesse dessa imagem de “entrada”, e posteriormente compara com as imagens do conjunto de treinamento. Para a Figura 34 o algoritmo irá classificar como *tricogaster*, devido a quantidade de pontos de interesse que foram detectados na imagem de treinamento.

A comparação não pode ser feita de forma igual ou diferente, pois é muito difícil encontrar pontos exatamente iguais, para isso utiliza-se algumas técnicas de busca aproximada como por exemplo, distância euclidiana, distância de mahalanobis, distância de cosseno, e muitas outras. Na literatura foram encontradas duas técnicas que já possuíam implementações disponíveis no OpenCV, FLANN (*Fast Library for Approximate Nearest*

Neighbors) e Força Bruta, que serão descritas abaixo.

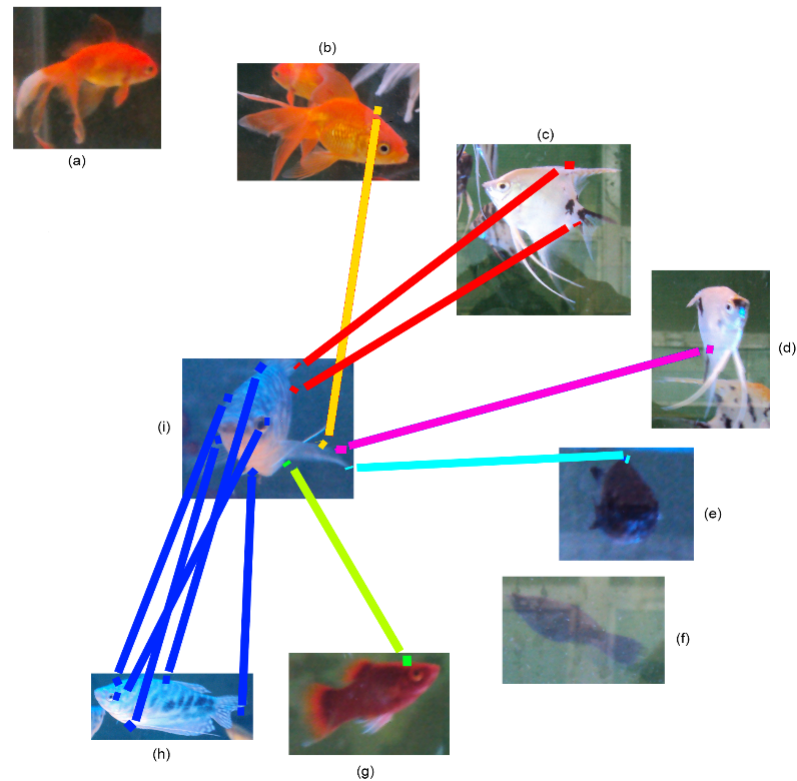


Figura 34: (a) e (b) espécie kingiuo. (c) e (d) espécie acará bandeira. (e) e (f) espécie molinésia preta. (g) platy sangue. (h) e (i) tricogaster azul. De (a) até (h) corresponde ao conjunto de treinamento e (i) seria a imagem de “entrada”. (a) e (f) nenhum ponto de interesse detectado. (b) (d) (e) e (g) foi detectado 1 ponto de interesse. (c) dois pontos de interesses detectados. (h) cinco pontos de interesse detectados. (i) onze pontos de interesse detectados

6.1 FLANN

Entre os diversos tipos de técnica para casamento de modelos encontrados na literatura, uma das utilizadas neste trabalho é conhecida como FLANN (*Fast Library for Approximate Nearest Neighbors*). Segundo Muja e Lowe (2009), FLANN é uma biblioteca para realizar pesquisas rápidas aproximadas, utilizando o método do vizinho mais próximo em espaços dimensionais grandes. O método do vizinho mais próximo (KNN - K-nearest Neighbors) consiste em atribuir uma classe a um elemento desconhecido observando a classe que predomina entre os seus vizinhos, utilizando uma determinada distância.

Conforme descreve Muja e Lowe (2009), FLANN possui um conjunto de algoritmos embutidos para realizar a busca por meio do vizinho mais próximo, além de contar também com um sistema para escolher automaticamente o melhor algoritmo e os parâmetros ideais de acordo com o conjunto de dados fornecido ao algoritmo.

Ainda segundo Muja e Lowe (2009), utilizando o algoritmo FLANN é possível atingir um alto desempenho para encontrar os vizinhos mais próximos utilizando apenas uma biblioteca. O autor cita ainda, que além da vantagem de ter um conjunto de algoritmos reunidos em um só, caso apareça novos algoritmos é fácil incorporar dentro do FLANN.

6.2 Força Bruta

O algoritmo de força bruta (ou *BruteForce*) possui várias áreas em que pode ser aplicado, possui várias áreas de aplicação, entre elas a da descoberta de senhas de usuários. Considere que a senha de um determinado site, só possa ter números (entre 0 e 9) e o tamanho deve ser 4, o método consiste em tentar todas as possibilidades existentes (0000, 0001, 0002, ... , 9999), e caso encontre a senha correta o processo é interrompido.

Esse método também aplica-se para o casamento de modelos utilizando imagens, mas com a principal desvantagem do alto custo computacional. Dada uma imagem de “entrada” esse algoritmo busca o descritor mais próximo na segunda imagem, ou seja, os pontos de interesse da imagem de “entrada” serão comparados com todos os pontos de interesse de cada imagem do conjunto de treinamento, realizando a busca por pontos de interesse semelhantes. Toda essa busca, dependendo do tamanho do banco de imagens, pode causar um alto custo computacional, devido ao número elevado de comparações que é feito.

7 Metodologia

Devido à grande variedade de espécies de peixes ornamentais disponível no mercado, foi realizado uma pesquisa em dois aquários da cidade de Campo Grande (MS), para verificar quais seriam as espécies mais comuns dentre as disponíveis no mercado, e como descrito no Capítulo 4 foram relacionadas as principais características de cada espécie, bem como uma breve descrição do ambiente em que deve ser criada.

O aplicação foi desenvolvida utilizando ubuntu 12.04, com o auxílio da biblioteca para Visão Computacional *OpenCV* versão 2.4.4¹. A linguagem de desenvolvimento foi C. Para auxiliar no desenvolvimento foi utilizada a IDE NetBeans ². A metodologia de desenvolvimento adotada foi o *SCRUM*³ com todo material produzido sob controle de versão do SubVersion⁴.

Na sessão 7.1 será explicado detalhadamente como foi o processo de construção do banco de imagens, bem como o local onde estará disponível. A sessão 7.2 elucidará como foi o processo de desenvolvimento do trabalho e obtenção dos resultados.

7.1 Banco de imagens

Devido a dificuldade em encontrar banco de imagens que possuíam as características necessárias para esse trabalho, foi realizado um levantamento sobre os locais que possivelmente poderiam ceder as imagens, e o lugar com a maior variedade de espécies de peixes foi a Flor de Liz, localizado em Campo Grande (MS). Gentilmente o proprietário do estabelecimento que está localizado na Av. Fernando Correa da Costa, 1045 - Centro em Campo

¹OpenCV é um pacote de bibliotecas que auxiliam no desenvolvimento de aplicações ligadas a Visão Computacional, pode ser encontrada em: <http://opencv.willowgarage.com/wiki/Welcome/Introduction>.

²Disponível em: netbeans.org/

³Metodologia ágil de desenvolvimento de software.

⁴O subversion é apresentado em <http://subversion.apache.org/>.

Grande (MS)⁵ cedeu a oportunidade de realizar a captura das imagens. As imagens foram capturadas em dois dias, dia 01/02/2013 das 13:30 às 15:30 e 15/02/13 das 09:00 às 10:00.

O local possui um grande acervo de peixes ornamentais, contando com mais ou menos 25 espécies distintas. Geralmente as espécies estão separadas em diferentes aquários, com diferentes tipos de iluminação e diferentes formas de injeção de oxigênio. A Figura 35 mostra o ambiente de captura, e como é possível observar no lado direito e no lado esquerdo ficam os aquários que possuem peixes e no fundo da imagem alguns aquários desativados⁶.

O ambiente possui uma iluminação geral que abrange todo o ambiente onde os aquários estão localizados, além disto, cada aquário possui uma iluminação específica dependendo da espécie de peixe que ali está localizada. A espessura do vidro é em torno de 5mm, devido a quantidade de água em cada aquário (aproximadamente 100 litros). Em todos os aquários que possuem peixes o estado do vidro está em boas condições, porém são encontrados aquários inativados pelo fato dos vidros não estarem nas melhores condições.



Figura 35: Ambiente de Captura

Na captura das imagens para a primeira versão do banco de imagens, foi utilizada a câmera Sony 14.1MP modelo DSC-W610⁷, com uma distância média de 10cm entre a câmera e o aquário e inicialmente sem zoom, porém devido a quantidade de peixes em alguns aquários foi dado zoom de 2x, para conseguir uma melhor imagem da espécie.

Para realizar a captura das imagens foi colocado a frente do aquário uma câmera, e a

⁵Google maps: <https://maps.google.com.br/maps?q=-20.46832,-54.611683>

⁶Mais fotos do ambiente de captura acesse: <http://goo.gl/pN8vo>

⁷Mais especificações em <http://poweredbyu.sony.com.br/suporte-e-servicos/>

partir daí foram tiradas diversas fotos (cerca de 30 fotos para cada espécie). Como é possível observar na Figura 36, essas imagens tem muitos peixes, e inicialmente o trabalho realizou o reconhecimento de um único peixe por vez. Tendo em vista esse problema, a partir das imagens originais foram geradas imagens onde enquadravam-se apenas um peixe (Figura 37).



Figura 36: Exemplo de imagem capturada no seu formato original.

A partir das imagens originais (Figura 36), foi gerado um conjunto de imagens para cada espécie e esse conjunto tendo em média 35 imagens cada um (a figura 38 exibe algumas dessas imagens). Para cada espécie procurou obter-se imagens em todos os ângulos do peixe, tendo em vista que o reconhecimento de uma determinada espécie de peixe tem que ser todos os ângulos. A Figura 37 mostra as imagens já normalizadas, e que foram as imagens finais utilizadas para o trabalho⁸.

A grade abaixo (Figura 38) mostra algumas imagens disponíveis no banco de imagens, através delas é possível perceber o desafio que foi realizar o reconhecimento de diferentes espécies de peixes em diferentes ângulos.

7.2 Delineamento experimental

⁸As imagens normalizadas estão disponíveis em: <http://goo.gl/Qhb9A>

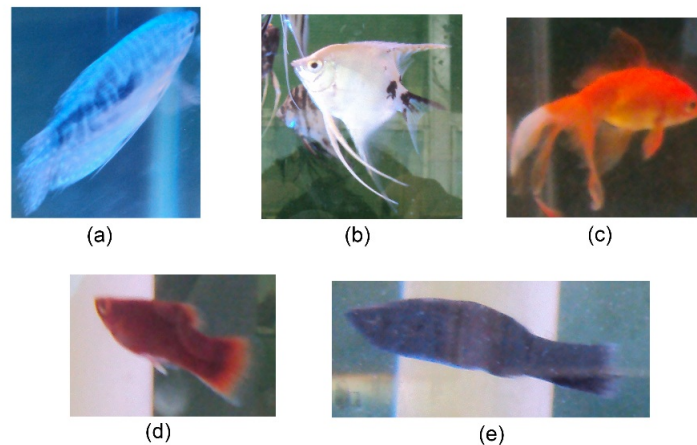


Figura 37: Cinco espécies de peixes que foram utilizadas no banco de imagens: Tricogaster Azul(a), Acará Bandeira(b), Kinguio(c), Platy Sangue(d) e Molinésia Preto(e)

O sistema desenvolvido realiza o reconhecimento de apenas um peixe por vez, podendo ser como entrada uma imagem já disponível ou uma imagem da webcam. Para esse primeiro momento a imagem que foi utilizada como “entrada” do algoritmo conteve apenas um peixe⁹. A Figura 39 mostra um exemplo de imagem que foi a “entrada” do algoritmo.

Para testar a solução adotada inicialmente, foram utilizados dois extratores de atributos e duas técnicas para casamento de modelos. Como opção de extratores de atributos foram utilizados o SURF e SIFT e para o casamento de modelos foi utilizado o FLANN e o Força Bruta. As seguintes combinações foram utilizadas:

1. SURF + FLANN
2. SURF + Força Bruta
3. SIFT + FLANN
4. SIFT + Força Bruta

A fim de obter um resultado médio estatístico, foram realizadas três repetições para cada combinação, alterando o conjunto de treinamento e de testes para cada uma das repetição, gerando três resultados distintos para cada métrica. A partir desses resultados, foram calculadas as médias para cada métrica e essas médias foram utilizadas para realizar o teste de Friedman¹⁰. Ao final de cada execução o *software* já salva os resultados daquela

⁹Há imagens que tem mais de um peixe, nesse caso o importante foi centralizar a imagem do peixe que deseja realizar o reconhecimento.

¹⁰Acesse o link <http://goo.gl/U7npW> e veja os resultados em cada uma das repetições



Figura 38: Grade contendo todas as espécie do banco de imagens em diferentes ângulos. Linha 1: Acará bandeira. Linha 2: Kinguio. Linha 3: Molinésia preta. Linha 4: Platy sangue. Linha 5: Tricogaster azul.

execução em um arquivo¹¹.

Para cada repetição, foram separadas aleatoriamente $\pm 70\%$ das imagens para treinamento e $\pm 30\%$ para teste dessa forma, em cada repetição realizada os conjuntos de treinamento e de testes eram distintos. O banco de imagens deste trabalho possui um

¹¹O arquivo chama results.txt e contém informações sobre a execução realizada, como data, extratores, resultado das métricas.



Figura 39: Exemplo de imagem para realizar o reconhecimento. Espécie de peixe *Platy sanguineus*. Imagem disponível no banco de imagens.

total de 229 imagens, sendo que para cada experimento foram utilizadas 158 imagens ($\pm 70\%$) para treinamento e 71 imagens ($\pm 30\%$) para testes¹². A proporção de separação foi mantida para cada classe, dessa forma uma determinada classe que possui 41 imagens, ao realizar a separação iria ficar 29 imagens ($\pm 70\%$) para treinamento e 12 imagens para ($\pm 30\%$) para teste.

Os algoritmos implementados foram comparados entre si utilizando o banco de imagens descrito anteriormente. Para cada algoritmo testado, foi calculado o desempenho médio referente às métricas precisão, medida-F, acurácia, taxa de verdadeiros positivos e taxa de falsos positivos. Através do software estatístico R¹³ versão 2.15.1, utilizando o teste de Friedman (FRIEDMAN, 1937), com cada bloco correspondendo a uma das classes dos problemas, foi identificado se os algoritmos testados diferem estatisticamente em relação ao desempenho. Para cada métrica, foi utilizado o menor valor p para realizar o teste *post-hoc*. O teste *post-hoc* também está disponível no R. Os *Box-plots* resultantes são possíveis visualizar no Capítulo 8.

¹²Acesse <https://www.dropbox.com/sh/hjlv4u48yejylh/YM3UeYCLPD> e veja os resultados para cada repetição.

¹³O software R está disponível em <http://www.r-project.org/>.

8 *Resultados, Análise e Discussão*

Neste capítulo, serão descritos os resultados e as análises que foram realizadas nesse trabalho. As métricas utilizadas nesse trabalho para medir estatisticamente a solução adotada inicialmente foram: taxa de verdadeiros positivos, taxa de falsos positivos, medida-F, acurácia e precisão. Abaixo seguem as tabelas e os diagramas de caixa referentes as métricas utilizadas nesse trabalho.

Tabela 1: Taxa de verdadeiros positivos

	SURF+FLANN	SURF+BRUTA	SIFT+FLANN	SIFT+BRUTA
Acará bandeira	0,44	0,43	0,37	0,34
Kinguio	0,53	0,67	0,72	0,78
Molinésia preta	0,42	0,67	0,62	0,79
Platy sangue	0,00	0,33	0,33	0,50
Tricogaster	0,34	0,32	0,57	0,58
Médias	0,34	0,48	0,52	0,60

Tabela 2: Taxa de falsos positivos

	SURF+FLANN	SURF+BRUTA	SIFT+FLANN	SIFT+BRUTA
Acará bandeira	0,16	0,15	0,11	0,11
Kinguio	0,16	0,15	0,15	0,14
Molinésia preta	0,19	0,19	0,16	0,15
Platy sangue	0,18	0,18	0,18	0,17
Tricogaster	0,04	0,06	0,07	0,06
Médias	0,15	0,14	0,13	0,13

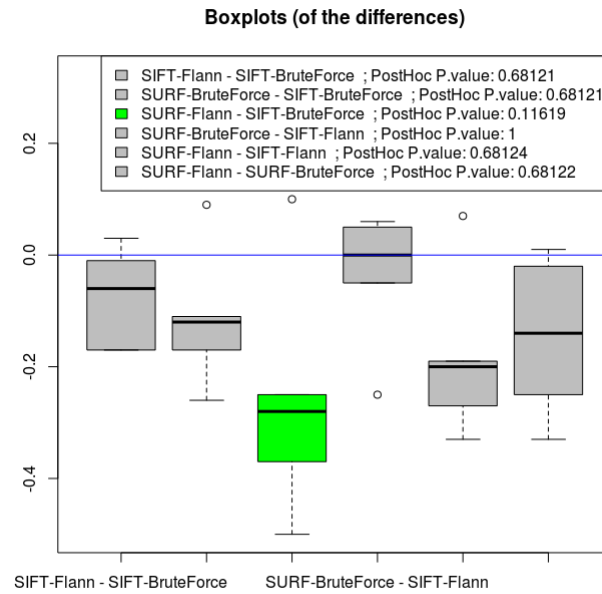


Figura 40: Resultado da análise *post hoc* para a métrica taxa de verdadeiros positivos

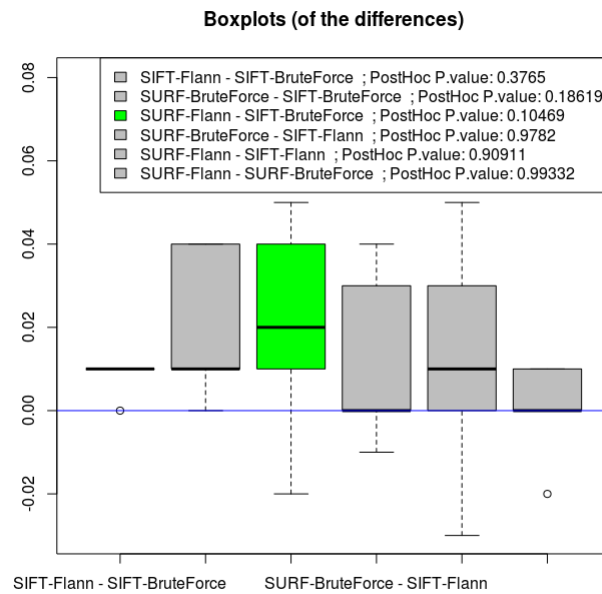
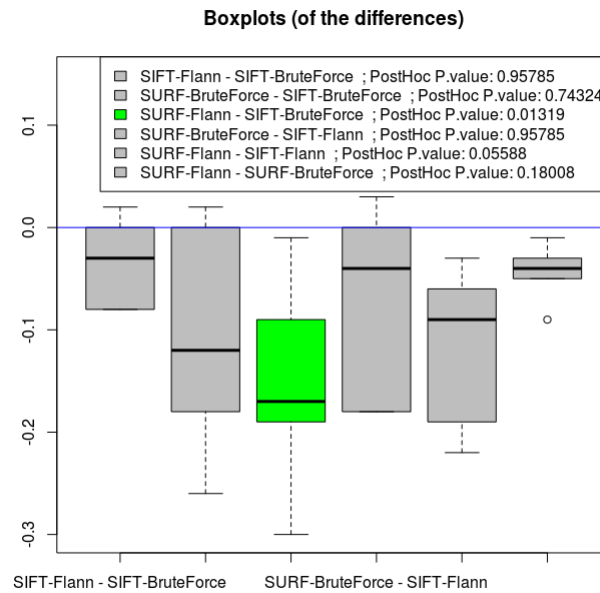


Figura 41: Resultado da análise *post hoc* para a métrica taxa de falsos positivos

De acordo com os resultados obtidos nas tabelas 1, 2, 3, 4 e 5 é possível observar que em todas as métricas utilizadas, a combinação SIFT+BRUTA obteve melhor resultado, com

Tabela 3: Medida-F

	SURF+FLANN	SURF+BRUTA	SIFT+FLANN	SIFT+BRUTA
Acará bandeira	0,48	0,51	0,51	0,49
Kinguio	0,29	0,38	0,35	0,38
Molinésia preta	0,12	0,16	0,34	0,42
Platy sangue	0,00	0,05	0,09	0,17
Tricogaster	0,45	0,46	0,64	0,64
Médias	0,27	0,31	0,39	0,42

Figura 42: Resultado da análise *post hoc* para a métrica medida-F

exceção da métrica taxa de falsos positivos que obteve um resultado médio igual ao da combinação SIFT+FLANN. Também é possível observar que as combinações SURF+BRUTA e SIFT+FLANN alcançaram um resultado bem próximo. Uma das possíveis causas é que essas combinações envolveram o extrator que obteve melhor resultado com a melhor técnica para casamento de modelos.

As Figuras 40, 41, 42, 43 e 44 mostram os diagramas de caixa resultantes da análise pós-teste para todas as métricas. Através dos diagramas de caixa é possível tirar diversas conclusões, entre elas a que as combinações SIFT+FLANN e SURF+BRUTA obtiveram uma diferença estatística, conforme é possível visualizar na marcação em verde. Além disso, também é possível visualizar que quando é utilizada a técnica de casamento de modelos de Força Bruta os algoritmos SIFT e SURF não apresentaram diferenças estatísticas. Isto também ocorre quando é aplicada a técnica de casamento de modelos FLANN.

Tabela 4: Acurácia

	SURF+FLANN	SURF+BRUTA	SIFT+FLANN	SIFT+BRUTA
Acará bandeira	0,70	0,70	0,59	0,55
Kinguio	0,82	0,83	0,84	0,85
Molinésia preta	0,79	0,81	0,82	0,85
Platy sangue	0,82	0,82	0,82	0,83
Tricogaster	0,59	0,63	0,83	0,85
Médias	0,74	0,76	0,78	0,79

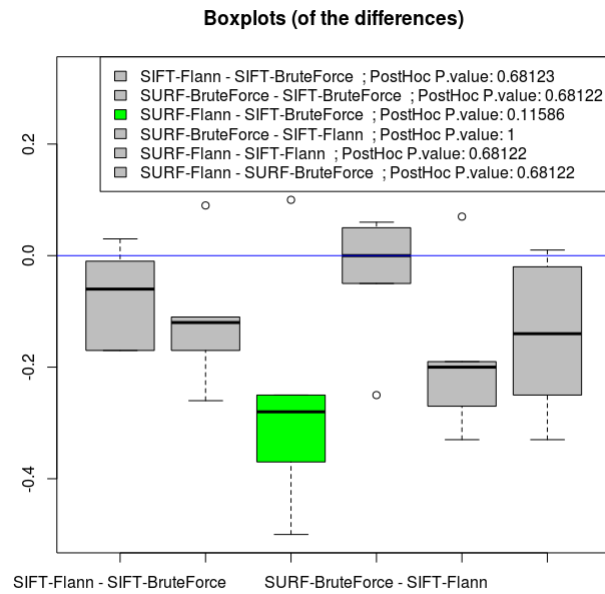
Figura 43: Resultado da análise *post hoc* para a métrica acurácia

Tabela 5: Precisão

	SURF+FLANN	SURF+BRUTA	SIFT+FLANN	SIFT+BRUTA
Acará bandeira	0,56	0,63	0,83	0,83
Kinguio	0,21	0,28	0,24	0,25
Molinésia preta	0,07	0,09	0,24	0,29
Platy sangue	0,00	0,03	0,05	0,10
Tricogaster	0,90	0,85	0,74	0,72
Médias	0,35	0,38	0,42	0,44

Para cada métrica foi encontrado o seu valor-p resultante do teste de Friedman. A Figura 40 exibe o diagrama de caixa referente a métrica taxa de verdadeiros positivos e o seu valor-p foi de 11,62%, já para a métrica taxa de falsos positivos o valor-p foi igual a 10,47% o que mostra a Figura 41. A Figura 42, representa os diagramas de caixa a métrica medida-F. Para a métrica medida-F o valor-p foi de 1,32%. As métricas acurá-

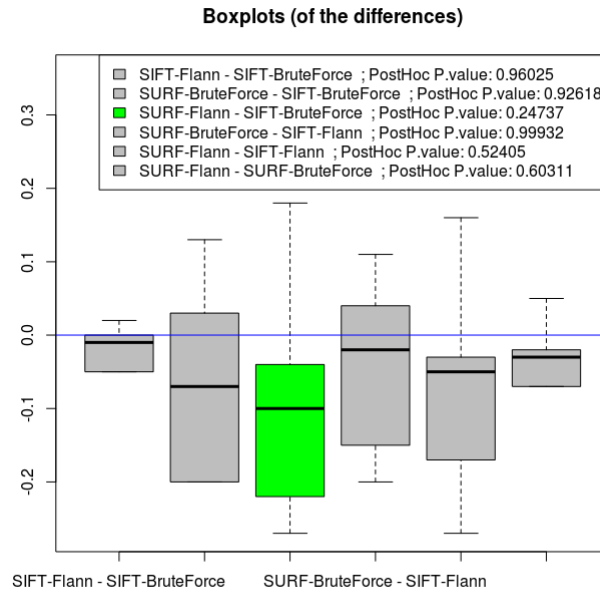


Figura 44: Resultado da análise *post hoc* para a métrica precisão

cia e precisão são obtidas logo abaixo. Para a métrica acurácia o valor-p encontrado foi 11,58% e para a métrica precisão foi utilizado o valor-p igual a 24,74%. Para todos os *box's plots* exibidos é possível visualizar que a combinação SIFT+FLANN obteve uma diferença estatística em relação ao SURF+BRUTA.

O melhor resultado alcançado pelo SIFT comparado ao SURF, deve-se a algumas etapas que o SIFT realiza com mais cautela em relação ao algoritmo SURF. Uma das etapas é busca por pontos de interesse, onde o algoritmo SURF realiza uma aproximação, tornando o algoritmo menos eficiente para o esse problema. Em relação a técnica para casamento de modelos é possível observar que o bruta força obteve um melhor resultado também, isso deve-se ao seu algoritmo que percorre todos os pontos de interesse de todas as imagens do conjunto de treinamento. Não houve uma diferença significativa entre o tempo de execução entre as diversas combinações.

É de se considerar que para as métricas taxa de verdadeiros positivos, medida-F e precisão a SURF+FLANN da espécie platy sangue alcançaram o resultado igual a zero. Talvez causado pelo seu formato semelhante a de outras espécie, como por exemplo o molinésia preta e o tricogaster, onde o algoritmo acaba confundido-se, e classificando erroneamente. Por outro lado é acreditável que o SIFT+BRUTA da mesma espécie para todas as métricas consegue classificar essa espécie, mesmo não possuindo um bom nível de exatidão.

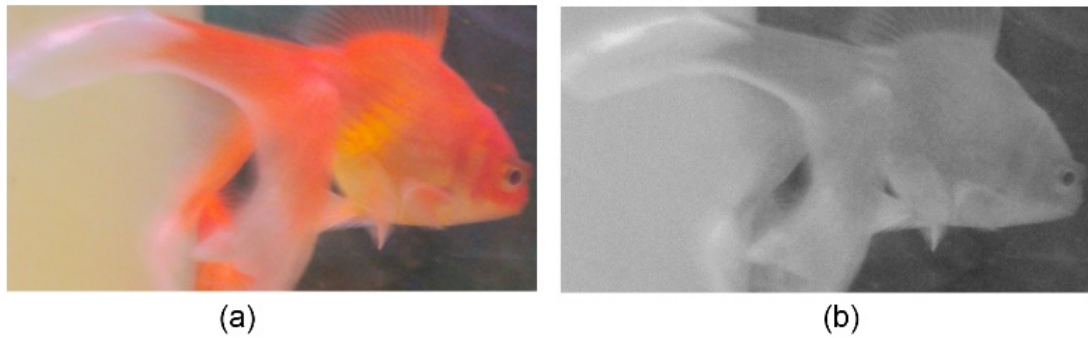


Figura 45: Peixe da espécie kinguio. (a) Imagem original e (b) imagem em tons de cinza

Como explicado no Capítulo 5 e exibido na Figura 45, o algoritmo SURF assim como o SIFT, transforma a imagem original para tons de cinza. Esse fato pode ser uma das possíveis causas para a baixa taxa de acerto do algoritmo (em torno de 41%). A cor pode ser considerada uma grande influenciadora na classificação de tais espécie, e utilizando algoritmos como o SURF e SIFT essas informações tem baixa importância, uma vez que a imagem original é transformada em tons de cinza (Figura 45 (b)).

Ainda na Figura 45, é possível visualizar que uma informação do fundo (um cano que alimenta o aquário de oxigênio), na imagem original (a), quando é transformada em tons de cinza (b) fica bem semelhante com a informação do peixe (nesse caso a cauda), ocasionando um provável erro do algoritmo.

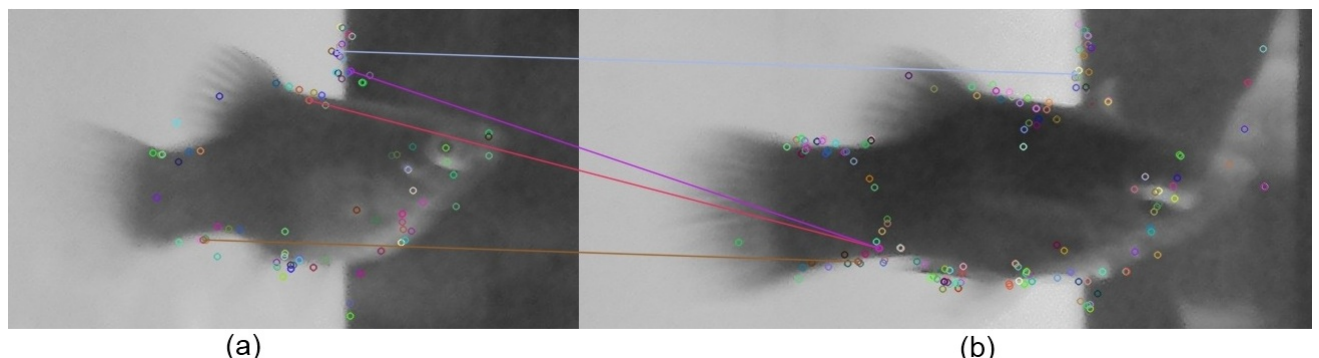


Figura 46: Espécie de peixe platy sangue. (a) representa a imagem que está sendo utilizada como entrada, e (b) a imagem comparada exibindo os pontos de interesse. A reta traçada entre os pontos de interesse de imagens distintas representa os pontos interessantes semelhantes nas duas imagens.

A Figura 46 exibe vários pontos de interesses detectados pelo algoritmo SURF. Como é possível visualizar na imagem citada há alguns pontos de interesse irrelevantes para o reconhecimento foram identificados. Esses pontos de interesse não deveriam influenciar na classificação de tal espécie e estão sendo utilizados pelo algoritmo, e muitas vezes estão influenciando diretamente na classificação de tal espécie. Um exemplo de informação que não deveria ser levada em consideração é o fundo do aquário e como é possível visualizar na Figura 46, essa informação foi detectada pelo algoritmo e utilizada para realizar a classificação.

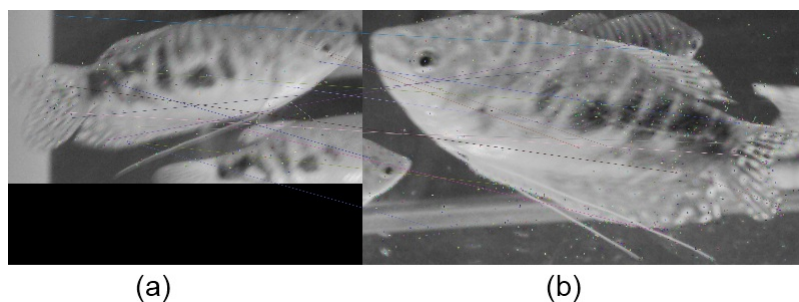


Figura 47: Espécie de peixe tricogaster. (a) representa a imagem que está sendo utilizada como entrada, e (b) a imagem comparada exibindo os pontos de interesse. A reta traçada entre os pontos de interesse de imagens distintas representa os pontos semelhantes nas duas imagens.

É importante salientar alguns pontos em que o algoritmo obteve um bom resultado. Na Figura 47 considerável avaliar os pontos de interesse detectados em ambas as figuras, e com isso observar que um conceito importante do algoritmo SURF foi atendido para esse problema. O conceito que se trata é o de cálculo de orientação dominante para invariância a rotação. Como visto na figura, alguns pontos de interesse da parte de trás do peixe foram detectados em ambas as figuras.

9 *Conclusões e Trabalhos Futuros*

A classificação de diferentes espécie de peixes permanece sendo um problema desafiador. Categorizar diferentes espécies de peixes em ambientes públicos, por exemplo o Aquário do Pantanal, é uma tarefa custosa devido a dificuldade que pessoas leigas têm para classificar visualmente as espécies de peixes. Um sistema de Visão Computacional foi proposto para resolver esse problema, e ajudar o público de forma geral a saber um pouco mais sobre as diferentes espécies que ali estarão presentes.

Para esse estudo, foi efetuado apenas um experimento, realizando algumas combinações entre os algoritmos para reconhecimento de padrões e os algoritmo de casamento de modelos. Foram executadas três repetições para cada combinação e extraídos os resultados das métricas taxa de verdadeiros positivos, taxa de falsos positivos, medida-F, acurácia e precisão. A repetição das combinações contribuiu para a minimização de possíveis desvios na obtenção dos resultados, e pudesse ser feita uma média entre os valores das métricas.

Com o experimento, foi possível observar que a combinação do algoritmo SIFT com o casamento de modelos baseado na Força bruta para a métrica de taxa de verdadeiros positivos obteve o melhor desempenho, em torno de 60%, seguido da combinação do SIFT com o FLANN, que foi de 52%. O resultado foi parcialmente satisfatório para o problema, porém alguns problemas foram identificados nos algoritmo de reconhecimento de padrões, um dos principais foi a não completa utilização da informação da cor do peixe. O teste de Friedman que foi aplicado nas métricas apontou uma diferença estatística entre as combinações SURF+FLANN e SIFT+BruteForce.

Como sugestão para trabalhos futuros, deverão ser buscados novos algoritmos para este problema e a realização de novos experimentos. Uma das possibilidades que será estudada é aplicar uma limiarização na imagem em que se deseja realizar o reconhecimento, tendo em vista obter uma imagem com a menor quantidade de ruídos. Uma das alternativas também é melhorar as imagens do conjunto de treinamento, recortando as imagens

e deixando apenas os peixes e retirando os ruídos existentes. Outra opção será utilizar extratores de atributos que aproveitam informações das cores dos peixes, devido a grande perda que se tem ao utilizar imagens em tons de cinza.

Também pretende-se, a partir dos resultados obtidos, desenvolver uma aplicação para dispositivos móveis nas plataformas mais utilizadas atualmente: iOS e android.

Referências

- ALSMADI, M. A. et al. Fish recognition based on robust features extraction from size and shape measurements using neural network. *Journal of Computer Science* 6 (10), p. 1059–1065, 2010.
- BAY, H. et al. Speeded-up robust features (surf). *Computer Vision and Image Understanding*, p. 110(3):346–359, Junho 2008.
- BRUNELLI, R. Template matching techniques in computer vision: Theory and practice the code companion. *Technologies of Vision*, 2009.
- CHAMBAH, M. et al. Underwater color constancy: enhancement of automatic live fish recognition. *SPIE / ISeT Electronic Imaging 2004*, San José, California, USA, jan 2004.
- FORSYTH, D.; PONCE, J. *Computer Vision: A Modern Approach*. [S.l.]: Prentice Hall, 2003. (Prentice Hall Series in Artificial Intelligence). ISBN 9780131911932.
- FRIEDMAN, M. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Journal of the American Statistical Association*, 1937.
- GRACIANO, A. B. V. *Rastreamento de objetos baseado em reconhecimento estrutural de padrões*. Dissertação (Mestrado) — São Paulo : Instituto de Matemática e Estatística, Universidade de São Paulo, 2007.
- HOLANDA, G. C. et al. Sistema de medição de distancia para robôs móveis utilizando algoritmo surf e visão estéreo. *V CONNEPI*, 2010.
- LEE, D.-J. et al. Contour matching for a fish recognition and migration-monitoring system. *Two - and Three-Dimensional Vision Systems for Inspection, Control, and Metrology II*, p. 37–48, 2004.
- LOWE, D. G. Scale-invariant feature transform (sift). *University of British Columbia, Vancouver, Canadá*, p. 28 p., 2004.
- MORAIS, B. D. S. Realidade aumentada em dispositivos móveis. *Universidade de Aveiro, Departamento de Electrónica, Telecomunicações e Informática*, 2011.
- MUJA, M.; LOWE, D. G. Fast approximate nearest neighbors with automatic algorithm configuration. In: *International Conference on Computer Vision Theory and Application VISSAPP'09*. [S.l.]: INSTICC Press, 2009. p. 331–340.
- NERY, M. S. et al. Determining the appropriate feature set for fish classification tasks. *Graphics, Patterns and Images, SIBGRAPI Conference on*, IEEE Computer Society, Los Alamitos, CA, USA, v. 0, p. 173–180, 2005. ISSN 1530-1834.

- OLIVEIRA, G. L.; ROCHA, C. H. Análise de detectores de características visuais em ambientes subaquáticos. *IV MCSUL - 4th Southern Conference on Computational Modelling, 2010, Rio Grande. Anais do 4 MCSUL*, 2010.
- QUINTA, L. N. B. *Visão Computacional aplicada na classificação de grãos de pólen*. Dissertação (Mestrado) — UCDB : Universidade Católica Dom Bosco, 2013.
- RACHID, L. C. C. L.; PEREIRA, A. A. S. Algoritmos de busca sift e surf no uso de dispositivos móveis. *II SEMULT - II Encontro Multidisciplinar da Unipac Ubá*, 2009.
- RIESENHUBER, M.; POGGIO, T. Models of object recognition. *Nature America*, Nature America, 2000.
- RIOS, L. R. S. Visão computacional. *Departamento de Ciência da computação - Universidade Federal da Bahia (UFBA) Salvador, Bahia, Brasil*, 2010.
- RODRIGUES, M. T. A. *Classificação Automática de Espécies de Peixes Baseada em Técnicas Robustas para Extração de Características e Sistemas Imunológicos Artificiais*. Dissertação (Mestrado) — Belo Horizonte : Centro Federal de Educação Tecnológica de Minas Gerais, 2009.
- SANTOS, R. M. dos. Um estudo de processamento de imagens com opencv. 2011.
- SCURI, A. E. Fundamentos da imagem digital. *Tecgraf/PUC-Rio*, 1999.
- SILVA, F. C. da. Análise roc. 2006.
- YANG, M.-H. Object recognition. *UCMERCED*, 2007.