

UM SISTEMA COLABORATIVO BASEADO EM INTELIGÊNCIA COLETIVA E RECONHECIMENTO DE OBJETOS DE LOCAIS PÚBLICOS E TURÍSTICOS

A COLLABORATIVE SYSTEM BASED ON COLLECTIVE INTELLIGENCE AND RECOGNITION OF OBJECTS FROM PUBLIC AND TOURISTIC PLACES

Adolfo Bocchi¹, Francisco Assis da Silva¹, Mário Augusto Pazoti¹, Danillo Roberto Pereira¹, Leandro Luiz de Almeida¹, Almir Olivette Artero²

¹Faculdade de Informática – FIPP, Universidade do Oeste Paulista – UNOESTE
E-mail: dolfao@gmail.com, {chico, mario, danilopereira, llalmeida}@unoeste.br

²Faculdade de Ciências e Tecnologia – FCT, Universidade Estadual Paulista – Unesp
e-mail: almir@fct.unesp.br

RESUMO - Este trabalho propõe um sistema colaborativo para favorecer aos consumidores de locais públicos e turísticos acesso à informação histórica e cultural sobre os objetos que compõe esses locais. Com esse sistema colaborativo é possível por meio de um aplicativo móvel obter a imagem de um objeto de um local público e turístico, e a enviar a um *WebService*, que realiza a detecção e reconhecimento, e compartilha as informações históricas e culturais referente ao objeto destacado na imagem. Para isso, o Sistema colaborativo faz uso do algoritmo SIFT e RANSAC e utiliza os conceitos da inteligência coletiva no compartilhamento das informações históricas e culturais. Foram realizados experimentos com imagens de objetos de locais públicos e turísticos para comprovar a metodologia proposta.

Palavras-chave: Inteligência Coletiva; Reconhecimento de Objetos; SIFT; RANSAC; Android.

ABSTRACT - This paper proposes a collaborative system for keeping the public and tourist sites of consumers access to historical and cultural information about the objects that make up these sites. With this collaborative system it is possible through a mobile application to get the image of an object of public and tourist site, and send to a *WebService*, which performs detection and recognition, and shares the historical and cultural information related to the highlighted object in the picture. For this, the collaborative system uses the algorithm SIFT and RANSAC and uses the concepts of collective intelligence sharing in the historical and cultural information. Experiments were performed with images of public and tourist sites objects to test the proposed methodology.

Keywords Collective Intelligence; Object Recognition; SIFT; RANSAC; Android.

Recebido em: 05/03/2015
Revisado em: 25/05/2015
Aprovado em: 30/06/2015

1 INTRODUÇÃO

Com a evolução dos meios de comunicação, em especial a Internet e os Smartphones, o acesso à informação ficou muito mais fácil, alterando a maneira como os consumidores realizam a prática do turismo. Ao longo dos anos, surgiram as redes sociais e aplicativos para facilitar as pessoas encontrar informações de trânsito, de transportes e de locais desejados.

Com a evolução dos meios de comunicação surgiu o *Crowdsourcing*, método que se utiliza da inteligência para resolver problemas, criar conteúdo, soluções ou desenvolver tecnologias (HOWE, 2009). Dessa forma, uma solução capaz de facilitar o acesso dos consumidores à informação, por meio de uma imagem de um local turístico, por exemplo, torna-se uma ferramenta de grande valia para *crowdsourcing*.

Visão Computacional é uma subárea da Ciência da Computação que tem por finalidade possibilitar que imagens sejam interpretadas por sistemas artificiais implementados em computadores a fim de que objetos sejam reconhecidos (MAIA, 2010; TREIBER, 2010).

Dentre muitas técnicas conhecidas na literatura, podem-se usar descritores locais para a atividade de reconhecimentos de objetos. Em geral, o reconhecimento e a descrição de objetos são realizados em duas etapas. A primeira consiste na utilização de

um detector de pontos chave e, em seguida, com a utilização de um descritor, gerar valores ou atributos suficientes para descrever esses pontos (SILVA et al., 2013).

Trabalhos recentes na literatura apresentam algoritmos que englobam diversas metodologias relacionadas aos processos de identificação de pontos chave, geração de descritores e correspondência entre os pontos. O algoritmo SIFT (*Scale Invariant Feature Transform*) (LOWE, 2004) identifica pontos estáveis de um objeto em uma imagem para extração dos descritores.

O algoritmo SIFT foi escolhido para ser utilizado neste trabalho, como parte do processo de reconhecimento dos objetos em imagens de locais públicos e turísticos, pela eficiência demonstrada em diversos trabalhos (SILVA, 2012; ALMEIDA, 2013; RODRIGUES, 2014).

A proposta deste trabalho é apresentar um sistema colaborativo que faz uso dos algoritmos SIFT e RANSAC aliados aos conceitos da inteligência coletiva e a tecnologia dos *Smartphones* para favorecer consumidores em locais públicos e turísticos. Dessa forma, este trabalho possibilita a um usuário obter informações históricas e culturais de locais, a partir da imagem de monumentos, estátuas e obras de arte.

Isso vem de encontro com a prática do turismo ou visitas em locais públicos em que as pessoas consomem cada elemento

que compõe aquele local, não apenas material, mas também do conhecimento histórico e cultural relacionado. Segundo dados do Ministério do Turismo, cerca de 100 milhões de pessoas praticaram o turismo no Brasil no ano de 2013 (BRASIL, 2013).

A partir de um estudo que reuniu informações que revelam o perfil dos consumidores de turismo, produzido pelo SEBRAE em parceria com a Confederação Nacional do Turismo (CNTUR), foi revelado que um dos pontos negativos e não tolerável é a falta de informações ou informações incorretas sobre o lugar, atrativos, localização, horários e, ainda, que os consumidores de turismo não abrem mão de acesso fácil e rápido à informação no destino turístico (SEBRAE, 2012).

O presente trabalho encontra-se organizado da seguinte maneira: a Seção 2 apresenta os conceitos da inteligência coletiva e as técnicas utilizadas para detecção e reconhecimento de objetos em imagens; na Seção 3 são apresentados os trabalhos relacionados, que serviram de motivação e contribuíram para a realização deste; na Seção 4 é apresentada a metodologia utilizada no desenvolvimento do sistema colaborativo; na Seção 5 são apresentados os experimentos utilizados para avaliar a eficiência da metodologia implementada, e por fim, na Seção 6 são apresentadas as considerações finais.

2 TRABALHOS RELACIONADOS

Nesta seção resumem-se as investigações realizadas para se obter conhecimento de trabalhos relacionados ao apresentado neste artigo, envolvendo inclusive abordagens diferentes. Esta busca objetivou o conhecimento para a aplicação dos conceitos da inteligência coletiva em aplicativos e sistemas colaborativos.

No trabalho de Lucio (2011), os conceitos da inteligência coletiva são aplicados com o objetivo de tornar o transporte público inteligente e facilitar o acesso de informações aos usuários. O autor aplica os conceitos da inteligência coletiva por meio de troca de mensagens enviadas em tempo real pelos usuários do transporte público sobre a situação da linha de ônibus que está sendo utilizado por estes usuários. Para isso, o autor desenvolveu um aplicativo para dispositivos móveis que consiste em obter a localização do usuário via GPS (*Global Positioning System*), triangulação de sinal GSM (*Global System for Mobile Communications*) ou a partir da sua localização inserida manualmente pelo usuário, para que, posteriormente, este possa enviar uma mensagem para o Twitter sobre o estado atual de determinada linha de ônibus. Estas informações ficam disponíveis para serem consultadas e compartilhadas por outros usuários a partir do aplicativo móvel.

O WikiCrimes (FURTADO et al., 2010) é um projeto desenvolvido pelo professor Vasco Furtado, da Universidade de Fortaleza, que compila ocorrências criminais, georreferenciadas utilizando recursos do Google Maps. Nele os usuários são responsáveis pela construção do conteúdo disponível, através de um wiki. Quando o usuário tem conhecimento de alguma ocorrência criminal, ele pode acessar o site WikiCrimes e registrar o fato no mapa. Assim, formam-se densidades de ocorrência de crimes, sendo possível observar onde há maior incidência de determinado delito.

Os conceitos da inteligência coletiva também estão aplicados no trabalho de Cruz et al. (2014), que por meio de um aplicativo móvel denominado Lista Fácil, auxilia o consumidor a fazer economia durante o processo de compras. O aplicativo móvel desenvolvido pelos autores permite ao consumidor obter uma noção da variação de preço antes que ele realize uma compra. Com o aplicativo, o consumidor cria listas de produtos, atualiza e compartilha preços, permitindo que outros consumidores tenham conhecimento sobre estas informações, e através do método simplex, realiza o cálculo de menor custo da lista de compras. No trabalho foi criado um *WebService* que realiza a integração entre o aplicativo móvel e o banco de dados, com comunicação baseada em XML (*eXtensible Markup*

Language). Para cada lista de produtos criada pelo usuário os dados são enviados pelo aplicativo móvel ao *WebService* que processa as informações, atualiza a base de dados e retorna o cálculo de menor custo da lista para o aplicativo.

3 REFERENCIAL TEÓRICO

Nesta seção são descritos os conceitos da inteligência coletiva e os algoritmos utilizados para a detecção e o reconhecimento de objetos em imagens, especificamente os algoritmos SIFT (LOWE, 2004) e RANSAC (FISCHLER; BOLLES, 1981).

3.1 Conceitos da Inteligência Coletiva

O uso dos *Smartphones*, “telefones inteligentes”, que são híbridos entre celular e computador e englobam algumas das principais tecnologias de comunicação: Internet, GPS, e-mail, SMS (*Short Message Service*) (TECHTUDO, 2011), acabou desenvolvendo novos hábitos na população que o utiliza, não apenas para se comunicar por meio de voz, mas também para enviar mensagens de texto, fotografar, pesquisar e acessar conteúdos na Internet, jogar, ouvir música, se orientar, localizar endereços utilizando dados de GPS (BUENO, 2013).

Utilizando a Internet como meio, a câmera do *Smartphone*, as mensagens de texto, juntamente à mobilidade, é possível produzir e distribuir informação em tempo

real. Algo acontece e instantaneamente já pode ser registrado por alguém e distribuído de modo que diversas outras pessoas, que tenham acesso à tecnologia fiquem sabendo do ocorrido em minutos (HOWE, 2009).

Em seus estudos, Lévy (2007) afirma que o fato de se poder trocar informações, de interagir com outras pessoas pelos meios de comunicação, traz um aprendizado recíproco. Ele define essa interação como Inteligência Coletiva, a qual consiste em uma inteligência distribuída em toda parte, incessantemente valorizada, coordenada em tempo real, que resulta em uma mobilização efetiva das competências.

O *Crowdsourcing* é um modelo de solução compartilhada de problemas através de uma rede de solucionadores, que normalmente se encontram em forma de comunidades que possuem um interesse em comum. Os problemas são transmitidos ao grupo em forma de convite e estes propõem soluções para estes problemas e também classificam outras soluções procurando as melhores (HOWE, 2009).

3.2 Técnicas para Detecção de objetos em Imagens

Lowe (1999) desenvolveu o algoritmo SIFT, que permite a detecção e extração de descritores para correspondência (*matching*) de diferentes imagens de um objeto ou cena. Esses descritores são razoavelmente invariantes à mudança de rotação e escala e

parcialmente invariante à iluminação e às projeções 3D.

O SIFT é um algoritmo dividido em duas partes: detector e descritor. O detector é baseado no cálculo de diferenças gaussianas e o descritor utiliza histogramas de gradientes orientados para descrever a vizinhança dos pontos de interesse. Os descritores que o algoritmo SIFT fornece são bem localizados, reduzindo assim a possibilidade de não haver correspondência (LOWE, 1999).

A obtenção de descritores de uma imagem pelo algoritmo SIFT é realizada por meio de quatro estágios principais. Os dois primeiros estágios descrevem a parte do detector e os dois seguintes a formação do descritor (LOWE, 2004).

O estágio de detecção de extremos no espaço escala representa a detecção dos pontos chave a partir da busca de características estáveis que identificam candidatos invariantes a escala e orientação. Isso é feito utilizando-se a função espaço escala (WITKIN, 1983). No algoritmo SIFT é usada a função Gaussiana como sendo o núcleo (*kernel*) dessa função espaço escala. De maneira sucinta, são detectados extremos (máximos ou mínimos) em uma pirâmide da imagem convoluída com a função de diferença de filtros Gaussianos. O espaço escala de uma imagem é definido como uma função $L(x, y, \sigma)$, que é dada a partir da

convolução de uma Gaussiana $G(x, y, \sigma)$, com uma imagem de entrada $I(x, y)$, conforme a Equação 1:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad (1)$$

em que $*$ é a operação de convolução em x e y , e a função Gaussiana é dada por:

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

No filtro Gaussiano, o valor de σ representa o parâmetro que define o fator de suavização de imagem. Esse primeiro estágio realiza a busca em todas as escalas e locais de imagem. Isso é feito utilizando-se a diferença de filtros Gaussianos de modo a identificar pontos de interesse invariáveis a escala e orientação. Este é o estágio mais custoso computacionalmente do algoritmo (LOWE, 2004).

Foi proposta por Lowe (1999) uma forma eficiente para detectar pontos chave estáveis usando os extremos do espaço escala com o uso de uma função de Diferença de Gaussianas (DoG – *Difference of Gaussian*). A diferença de Gaussianas, $D(x, y, \sigma)$, é computada pela diferença de duas imagens filtradas em escalas próximas, separadas por uma constante multiplicadora k . A função DoG é definida pela Equação 3:

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \quad (3) \\ = L(x, y, k\sigma) - L(x, y, \sigma)$$

Esta convolução é a diferença entre imagens suavizadas por um filtro Gaussiano em escalas σ e $k\sigma$. A função *DoG* consegue detectar variações de intensidade na

imagem, elimina detalhes indesejados e realça características fortes.

O segundo estágio é a localização de pontos chave para cada local candidato, em que é detectado um extremo (máximo ou mínimo), e a partir disso, um modelo detalhado é ajustado para se determinar a localização e escala. Pontos chave, ou pontos de interesse, são então selecionados baseando-se em suas medidas de estabilidade; são rejeitados os pontos de baixo contraste e localizados ao longo de bordas.

Segundo Lowe (2004), todos os pontos detectados como extremos são possíveis candidatos a pontos chave, e o próximo passo é executar um ajuste detalhado aos pixels próximos em relação a escala, localização e relação de curvaturas principais.

No estágio de determinação de orientação, uma ou mais orientações são associadas para cada imagem. Todas as operações seguintes são realizadas nos dados da imagem relativamente transformados em relação à orientação, escala e localização de cada ponto chave, assim provendo a invariância a essas transformações (MAIA, 2010).

A escala do ponto chave é utilizada para selecionar a imagem suavizada L com o filtro Gaussiano. Dessa maneira, todos os cálculos passam a ser executados com

invariâncias à escala. Para cada imagem $L(x, y, \sigma)$, na mesma escala σ , a magnitude do gradiente $m(x, y)$ e a orientação $\theta(x, y)$ são computadas usando diferenças de pixels, conforme as Equações 4 e 5.

$$m(x, y) = \sqrt{\Delta x^2 + \Delta y^2} \quad (4)$$

$$\theta(x, y) = \tan^{-1}(\Delta y/\Delta x) \quad (5)$$

sendo $\Delta x = L(x + 1, y) - L(x - 1, y)$ e $\Delta y = L(x, y + 1) - L(x, y - 1)$.

O quarto estágio é o de extração dos descritores dos pontos chave. Nesse estágio, os gradientes locais da imagem são medidos na escala selecionada, na região ao redor de cada ponto chave. Estas medidas são então transformadas para uma representação que permite observar níveis significantes de distorção de forma e mudança na iluminação. Em seguida são criados histogramas de orientação para compor o descritor (Figura 1).

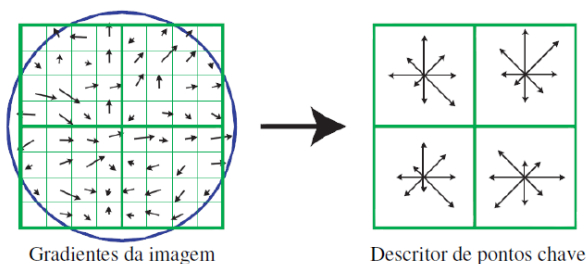


Figura 1. Gradiente da imagem e descritor de pontos chave.

Fonte: (LOWE, 2004).

O peso referente à magnitude de cada pixel é atenuado pela função Gaussiana. A função Gaussiana não é aplicada de modo idêntico ao do estágio anterior. No caso é utilizado um peso α para interpolar a direção relativa no histograma.

Para cada imagem são extraídos diversos descritores, cada um referente a um ponto chave. Quando se aplica o algoritmo SIFT em uma imagem, tem-se como resultado um conjunto de descritores, que são usados para se fazer a correspondência (*matching*) entre diferentes imagens.

Para a detecção e reconhecimento de objetos em imagens tomando como base a correspondência de pontos chave extraídos pelo algoritmo SIFT (LOWE, 2004), faz-se necessário eliminar os pontos que não correspondem, ou não se ajustam com o modelo de objeto desejado. São apenas preservados aqueles pontos que realmente condizem com o objeto desejado. Nesse caso, o algoritmo RANSAC (*RAN*dOm *SAM*ple *CON*sensus) (FISCHLER; BOLLES, 1981) é um dos mais comumente encontrados na literatura para a obtenção de um resultado robusto.

Este algoritmo é um método de estimação robusto projetado para extração dos *inliers*¹ e *outliers*² do conjunto de pontos chave. Tem sido muito usado para o reconhecimento de objetos (COLLET et al., 2009; OKABE; SATO, 2003), pois possibilita encontrar correspondências geometricamente consistentes para resolver

¹ *inliers*: pontos de dados que se ajustam com um determinado modelo desejado dentro de uma certa tolerância de erro.

² *outliers*: pontos de dados que não se ajustam ao modelo correspondente ao objeto desejado, estão fora de uma certa tolerância de erro.

o problema de junção de pares de imagens, mesmo em condições extremas, ou com algum tipo de *outlier*.

Ao contrário das técnicas convencionais que usam grande quantidade de dados para obter uma solução inicial, e em seguida eliminar os *outliers*, o RANSAC usa um conjunto com um número mínimo de pontos para uma primeira estimativa e continua o processo, aumentando o conjunto de pontos de dados consistentes (FISCHLER; BOLLES, 1981).

3.2 Árvore k-d Tree

A árvore k-d tree foi introduzida em Bentley (1975) como uma generalização de uma árvore binária de dimensão elevada.

Segundo Samet (2006), a k-d tree é uma árvore binária relacionada a coordenadas de k dimensões. Esta estrutura multidimensional testa, em cada nó percorrido, um valor chave que indica a dimensão utilizada na partição do espaço.

Esse valor subdivide os nós restantes nas subárvores esquerda e direita, sendo necessária apenas a análise de uma coordenada a cada visita de um nó. Cada nível da árvore compartilha o mesmo valor chave. Ao longo dos níveis da árvore, os valores chave variam por alternância circular.

Lima, Mota e Pinto (2012) definem o princípio de funcionamento de uma k-d tree, em que cada nível da árvore se refere a uma dada dimensão. Assim, todo nó obedece à seguinte propriedade: os pontos com valores da dimensão chave menores que o nó referência ficam na subárvore esquerda e os demais na subárvore direita. A dimensão que será utilizada como discriminador é estabelecida de forma alternada pelos diversos níveis da árvore binária. A Figura 2 apresenta uma k-d tree com $k = 2$ para 8 pontos. A Figura 2(a) mostra a maneira como o plano é subdividido, e a Figura 2(b) mostra a representação da árvore obtida.

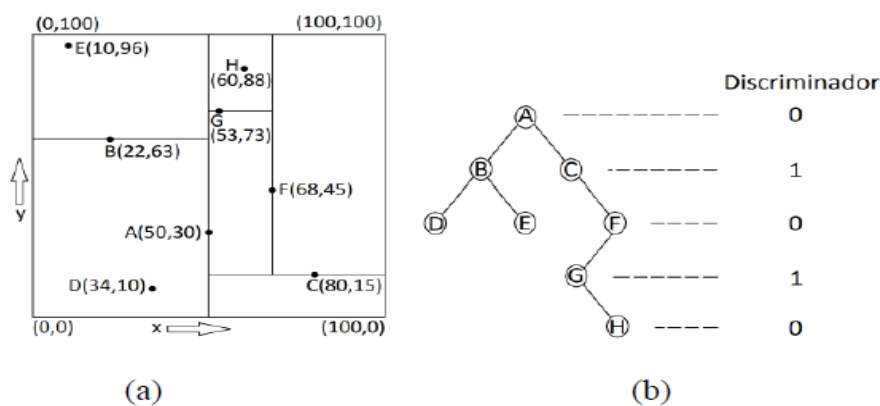


Figura 2. Exemplo da representação de uma k-d tree. Em (a) a maneira como o plano é subdividido, e em (b) a representação da árvore obtida.

Fonte: (LIMA; MOTA; PINTO, 2012).

4 METODOLOGIA

A implementação da metodologia proposta neste trabalho deu origem a um Sistema colaborativo denominado *WholsMonument*. O sistema é composto por um aplicativo móvel desenvolvido para Sistemas operacionais Android, um *WebService* desenvolvido na linguagem Java, que se comunica com uma base de dados do tipo relacional. O *WebService* é responsável por fazer a integração do aplicativo móvel com a Web e o banco de dados (Figura 3).

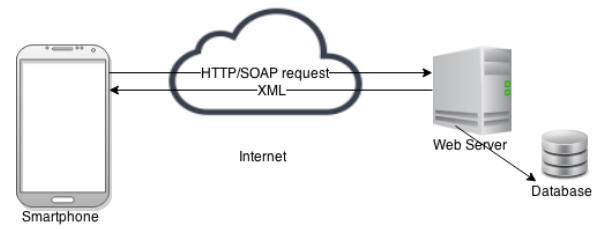


Figura 3. Integração e fluxo de dados entre o aplicativo e o *WebService*.

Para a implementação do *WebService* foi modelado e construído um banco de dados utilizando o MySQL 5.5.27 e o MySQL Workbench 6.2. A Figura 4 ilustra o Modelo Entidade Relacionamento do banco de dados construído.

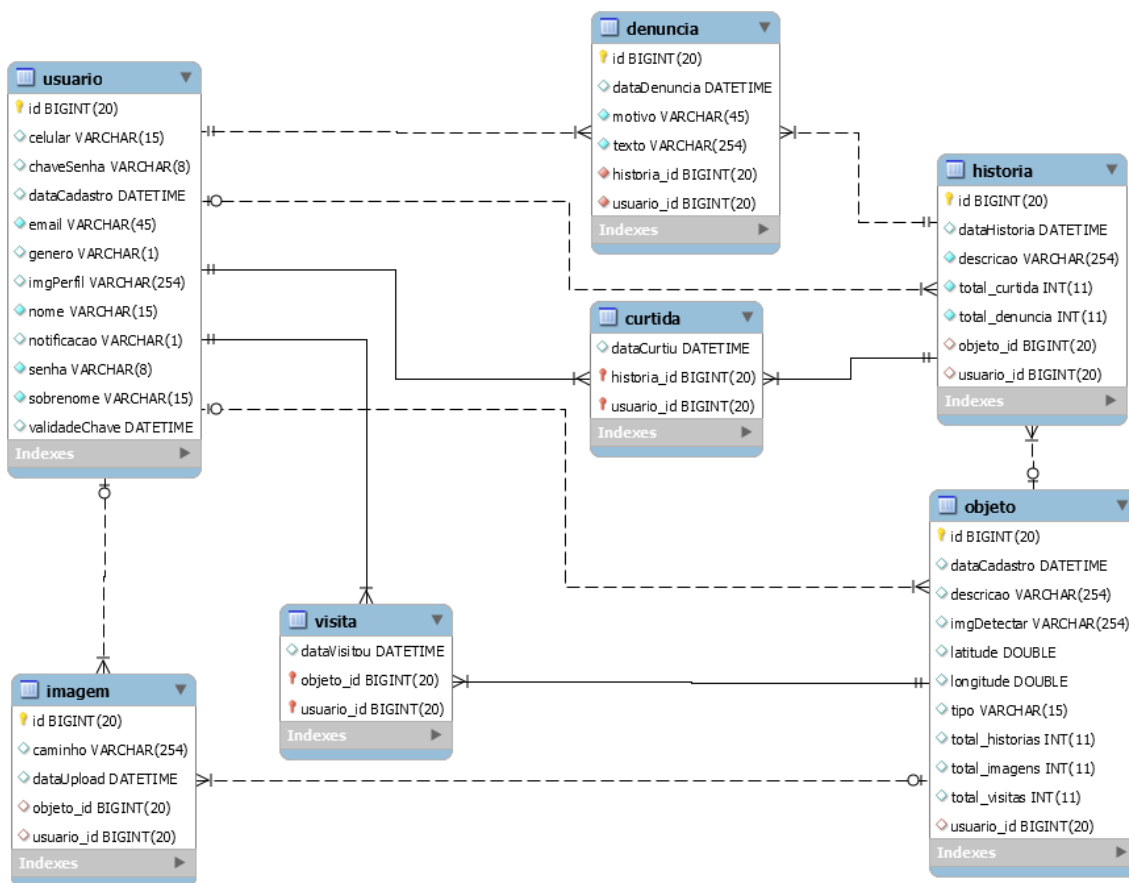


Figura 4. Modelo Entidade Relacionamento do banco de dados modelado para o Sistema colaborativo *WholsMonument*.

O modelo de dados apresentado na Figura 4 torna possível armazenar: dados sobre os usuários, para identificação e

validação de acesso; dados da imagem para detecção e reconhecimento do objeto, localização, descrição e tipo do objeto

adicionado pelo usuário; dados sobre histórias e imagens referente a um objeto, compartilhadas por um usuário; permite verificar a relevância de uma história por meio de curtidas ou de denúncias feitas pelos usuários; e as visitas feitas pelos usuários em objetos cadastrados.

Para que o *WebService* realize a persistência dos dados no banco de dados, e execute as funcionalidades do Sistema colaborativo *WholsMonument*, foi modelado um diagrama de classes para auxiliar no desenvolvimento do Sistema colaborativo.

Um diagrama de classes mostra um conjunto de classes, interfaces, colaborações e seus relacionamentos. São importantes para a visualização, a especificação e documentação de modelos estruturais. Estas classes foram modeladas através da Linguagem de Modelagem Unificada (UML – *Unified Modeling Language*) (BOOCH et al., 2006) e implementadas utilizando a linguagem de programação orientada a objetos Java.

Nas classes apresentadas no diagrama (Figura 5) estão codificadas todas as funcionalidades do sistema e para uma melhor organização elas foram agrupadas segundo suas funcionalidades em pacotes (*packages*), para cada parte do *WebService* do Sistema colaborativo *WholsMonument*.

A Figura 5 ilustra o diagrama de classes simplificado dos pacotes do *WebService*.

A Figura 6 apresenta o diagrama de classes simplificado dos pacotes “logica” e “util”. As classes do pacote “logica” implementam uma estrutura de dados de uma árvore k-d tree utilizada na inicialização do *WebService* para carregar os objetos cadastrados em memória e favorecer a busca de um objeto, durante o estágio de detecção e reconhecimento de um monumento por meio de uma imagem de um local público e turístico. O pacote “util” contém a classe “Sift” que implementa os códigos para utilização do algoritmo SIFT na linguagem Java. Os métodos da classe “Sift” fazem uso das funções da biblioteca de Visão Computacional OpenCV (OPENCV, 2014) a partir de seus arquivos .DLL (*Dynamic Link Library*). A classe “Funcoes” possui métodos para formatar datas, salvar um *array* de bytes em um arquivo de imagem, converter um arquivo de imagem em um *array* de bytes e um método que realiza cálculo da distância entre dois pontos. Os métodos para o envio de e-mail estão nas classes “Smtplib” e “EnviaEmail”, o sistema envia e-mails para os usuários quando ocorre uma denúncia em uma história, quando o usuário deseja recuperar sua senha ou quando o usuário confirma seu cadastro no aplicativo.

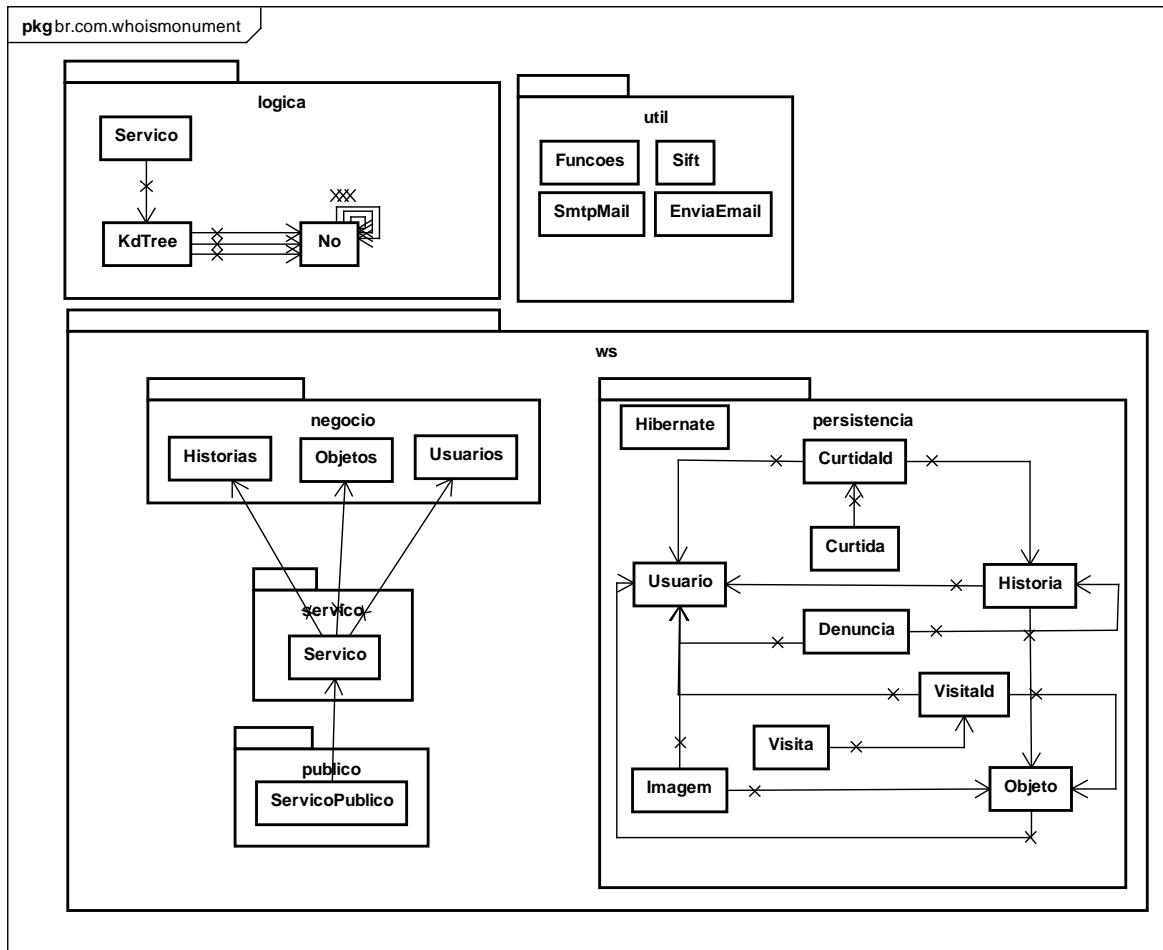


Figura 5. Diagrama de classes simplificado do WebService.

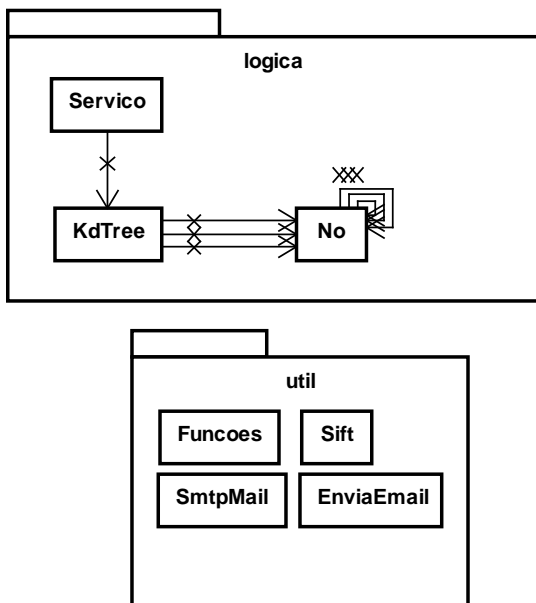


Figura 6. Classes do pacote “logica” e do pacote “útil” com suas relações.

As classes do pacote “negocio” e “servico” são apresentadas no diagrama simplificado e ilustrado na Figura 7.

As classes “Historias”, “Objetos” e “Usuarios” do pacote “negocio” são responsáveis por acessar o banco de dados do Sistema colaborativo *WholsMonument*. Todas essas classes fazem uso das classes do pacote “persistencia” mostrado na Figura 8, para realizar a inserção, consulta, alteração e exclusão do banco de dados. A classe “Servico” é comentada com anotações JAX-WS e define os métodos que ficam disponíveis na Web, ao iniciar o *WebService*.

O *WebService* é iniciado e publicado, através da classe “ServicoPublico” no pacote “publico”, através do método “main”.

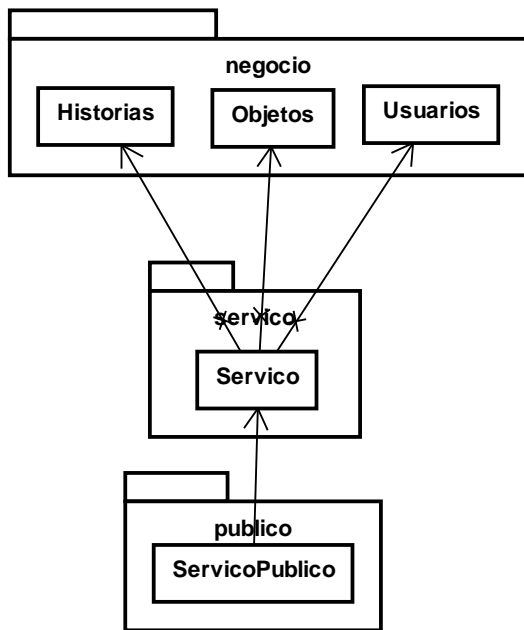


Figura 7. Classes do pacote “negocio” do pacote “servico” e do pacote “publico” com suas relações.

O *WebService* faz uso do Hibernate para realizar a persistência dos dados. O Hibernate é uma ferramenta de Mapeamento Objeto/Relacional. O termo Objeto/Relacional de Mapeamento (ORM) refere-se a técnica de mapeamento de dados com um esquema baseado na Linguagem SQL (*Structured Query Language*) (HIBERNATE, 2014).

As classes do pacote “persistencia”, ilustradas na Figura 8, implementam métodos e atributos que possuem anotações Hibernate que acessam as funções da biblioteca do Hibernate 4.3.6, a partir de seus arquivos .JAR.

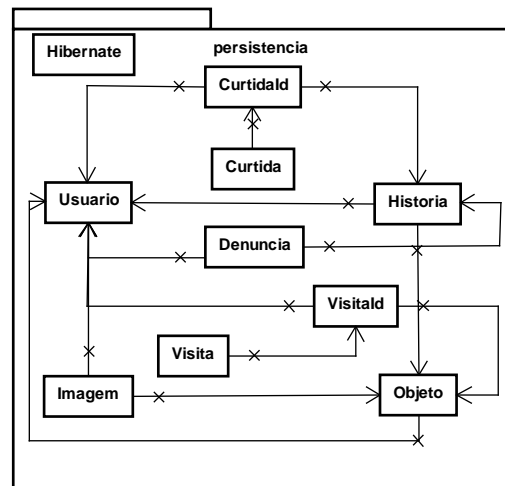


Figura 8. Classes do pacote “persistencia” com suas relações.

A funcionalidade principal do *WebService* é localizar, detectar e reconhecer um objeto a partir de uma imagem de um local público e turístico, e por meio da detecção do objeto permitir que o usuário compartilhe seu conhecimento histórico e cultural.

A metodologia utilizada para a detecção e reconhecimento de um objeto em imagem de um local público e turístico tem como uma de suas etapas o envio da imagem, latitude e longitude capturadas pelo aplicativo móvel. Para isso, o aplicativo móvel *WholsMonument* faz uso dos métodos nativos do Android para acionar o dispositivo de câmera integrada ao Smartphone, e capturar a imagem do objeto desejado. Para a captura das coordenadas de latitude e longitude, o aplicativo móvel faz uso dos métodos nativos do Android para acionar o dispositivo de GPS integrado ao Smartphone.

O aplicativo móvel, desenvolvido nesse trabalho, utiliza a biblioteca Ksoap2 versão 2.1.2 para gerar o envelope XML que consome os métodos do *WebService*. A biblioteca Ksoap2 tem como objetivo interagir com a maioria dos serviços que utilizam SOAP (*Simple Object Access Protocol*) como protocolo de comunicação. A KSOAP é

composta por um interpretador XML, uma camada de transporte e serialização. (KSOAP2, 2014).

A Figura 9 apresenta de forma resumida um fluxograma da metodologia utilizada para a detecção e reconhecimento de um objeto em uma imagem.

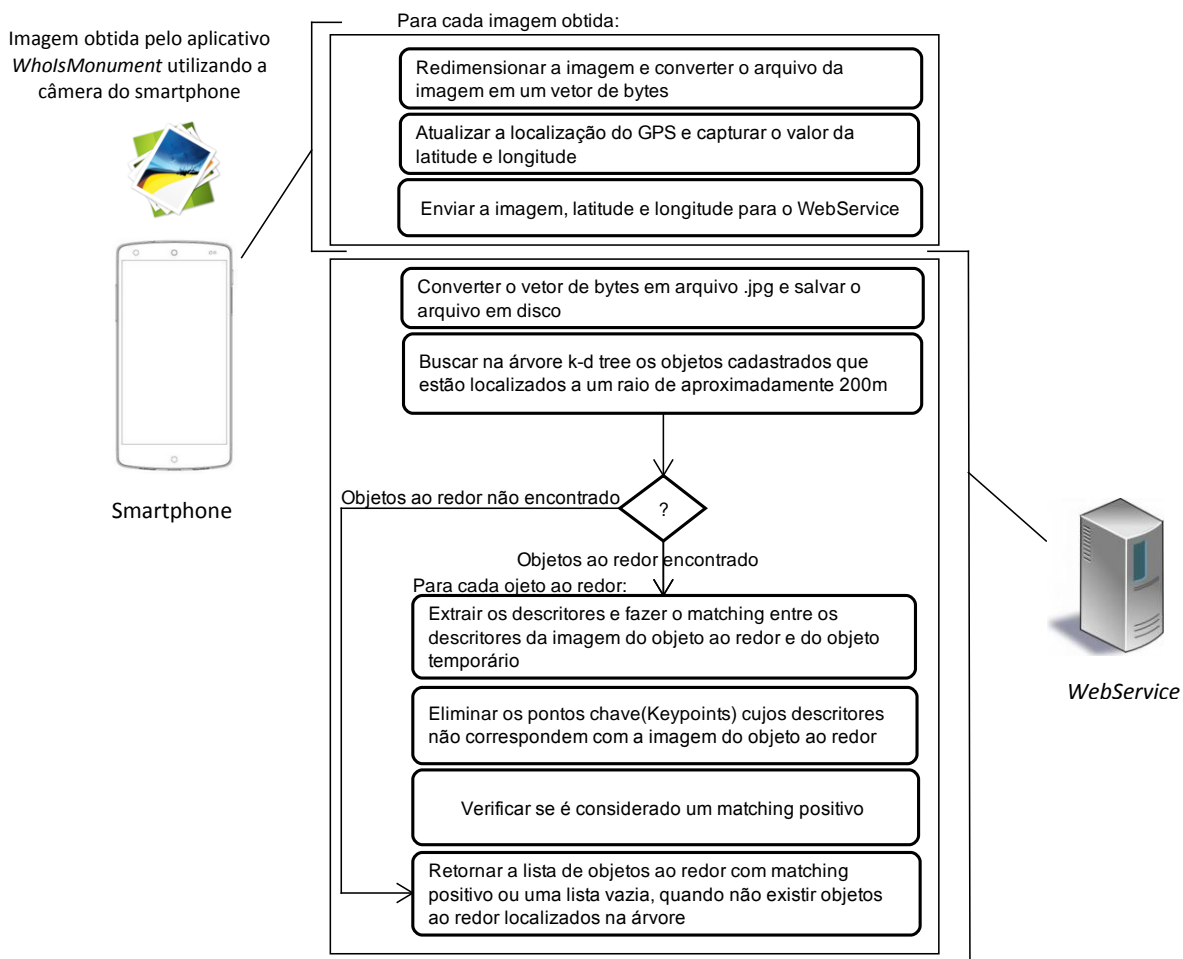


Figura 9. Fluxograma da metodologia utilizada para detecção e reconhecimento de um objeto em imagem de um local público e turístico.

O formato dos pacotes XML utilizado para a comunicação entre o aplicativo móvel e os métodos do *WebService* estão descritos na sequência. Na Figura 10(a) tem-se o formato do XML de recepção gerado pelo

método “getObjArredores” do *WebService*, e a Figura 10(b) o envelope XML criado pelo aplicativo móvel *WholsMonument* utilizando os métodos da biblioteca Ksoap2, contendo

as informações para consumir o método “getObjArredores” do *WebService*.

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope"/
xmlns:who="http://whoisonument_webservice/">
  <soapenv:Header/>
  <soapenv:Body>
    <who:getObjArredores>
      <img>cid:1131787538846</img>
      <latitude>?</latitude>
      <longitude>?</longitude>
      <usu_id>?</usu_id>
    </who:getArredores>
  </soapenv:Body>
</soapenv:Envelope>
```

(a)

```
<v:Envelope xmlns:i="http://www.w3.org/2001/XMLSchema-
instance"
  xmlns:d="http://www.w3.org/2001/XMLSchema"
  xmlns:c="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:v="http://schemas.xmlsoap.org/soap/envelope/">
  <v:Header />
  <v:Body>
    <n0:getObjArredores id="o0" c:root="1" xmlns:n0=
"http://whoisonument_webservice/">
      <img i:type="c:base64"> /9j/4AAQSk... </img>
      <latitude i:type="c:string">-22.123290</latitude>
      <longitude i:type="c:string">-51.387617</longitude>
      <usu_id i:type="d:long">2</usu_id>
    </n0:getArredores>
  </v:Body>
</v:Envelope>
```

(b)

Figura 10. Formato do XML para comunicação entre o aplicativo móvel e o *WebService*. Em (a) tem-se o formato do XML de recepção gerado pelo *WebService*, e em (b) o envelope XML gerado pelo aplicativo utilizando os métodos da biblioteca Ksoap2.

Após o método “getObjArredores” publicado no *WebService* ser consumido pelo aplicativo móvel (Figura 11), a detecção e reconhecimento do objeto na imagem enviada ao método “getObjArredores” é realizada pelo algoritmo implementado na metodologia desenvolvida neste trabalho e ilustrado de forma global na Figura 11. Esse diagrama de sequência é descrito de forma resumida a seguir.

A imagem enviada pelo aplicativo móvel é armazenada em um local temporário (método “saveImgTmp”). Essa imagem é usada pelo algoritmo de detecção e reconhecimento de objetos em imagens.

São buscados os objetos cadastrados, e armazenados na árvore k-d tree em memória, e que estão localizados a uma distância de no máximo 200 metros com

relação a latitude e longitude enviada pelo aplicativo móvel (“método buscaArredor”), e A implementação desta busca está descrita no Algoritmo 1.

Para cada objeto retornado pela busca são executados os estágios seguintes:

- Executar o algoritmo SIFT (LOWE, 2004) para extrair os descritores de pontos chave da imagem do objeto localizado ao redor e da imagem temporária;
- Executar o algoritmo para computar a correspondência (*matching*) entre os descritores dos pontos chave da imagem do objeto e da imagem temporária;
- Eliminar os pontos chave que possuem orientações que não condizem com o objeto cadastrado. Considera-se neste trabalho que os objetos possam ter apenas uma pequena rotação

(aproximadamente 5 graus) em relação a sua posição correta;

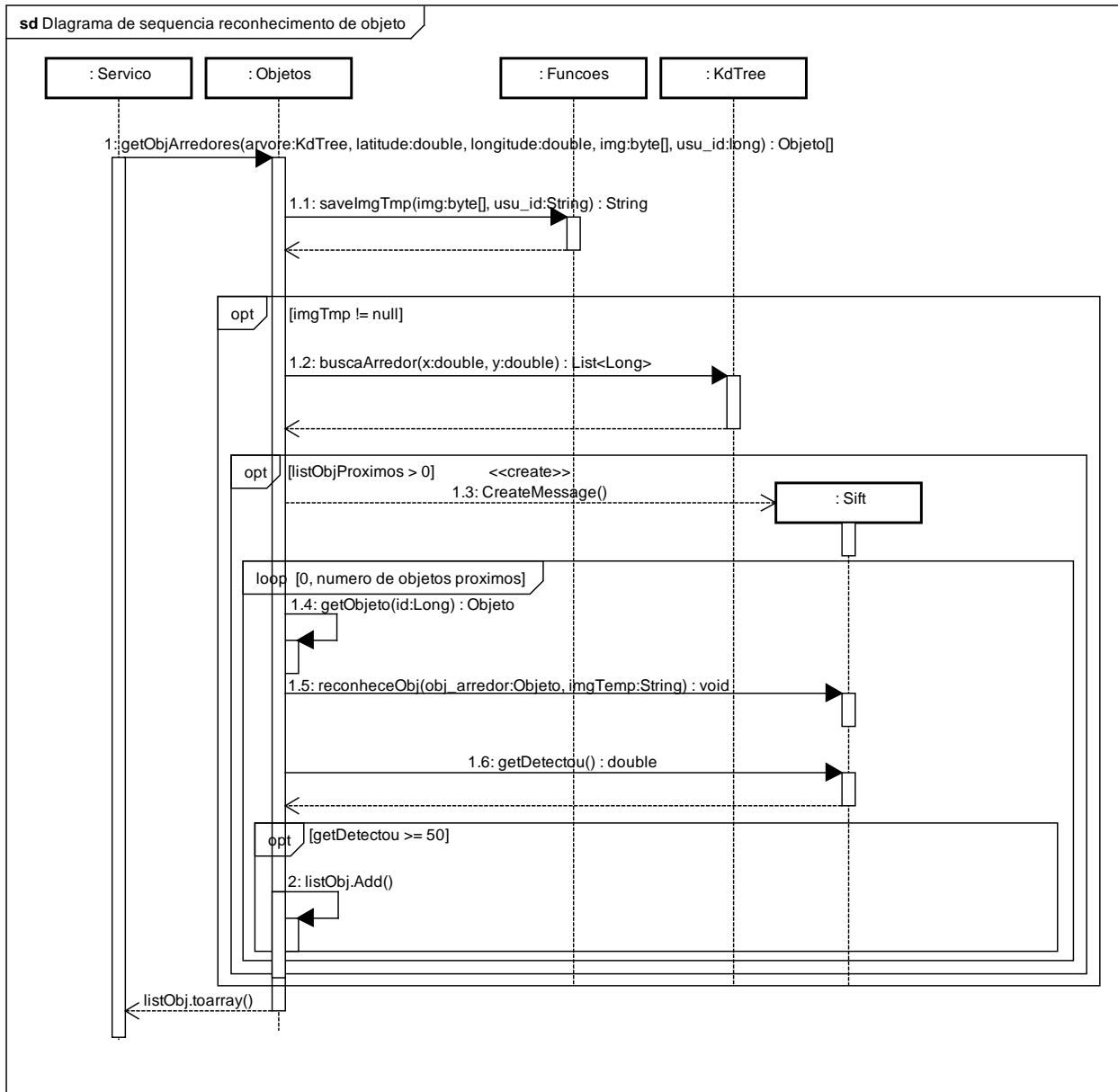


Figura 11. Diagrama de seqüência do algoritmo de detecção e reconhecimento de objeto em uma imagem de um local público e turístico.

- Executar o algoritmo para encontrar uma região aproximada do objeto cadastrado, em função dos pontos chave da imagem temporária, e após isto, os pontos chave que não pertencem a esta região são eliminados do conjunto de pontos prováveis a serem considerados *inliers*;
- Após eliminar os *outliers* (pontos que não correspondem com o objeto cadastrado), uma região retangular ao redor do objeto de interesse é encontrada;
- Calcular a porcentagem entre o total de *inliers* encontrado nas imagens e o total de *inliers* que estão dentro da margem de rotação (aproximadamente 5 graus) definida nessa metodologia;
- Adicionar o objeto cadastrado à lista de objetos detectados caso o resultado do

cálculo da porcentagem for maior ou igual a 50;

- Retornar a lista de objetos detectados após realizar o procedimento para todos os objetos da lista de objetos ao redor.

A busca de objetos cadastrados ao arredor do objeto desejado é parte da metodologia de detecção e reconhecimento de um objeto em imagem de um local público e turístico. O Algoritmo 1 implementa a busca na árvore k-d *tree* por objetos próximos à latitude e longitude associada à imagem do ponto turístico.

Algoritmo 1 – buscarProximos(raiz, latitude, longitude, listaRetorno)

Entrada: O ponto *raiz* da árvore k-d *tree*, a *latitude* e *longitude* do ponto atual e uma lista vazia de objetos *listaRetorno*.

Saída: Uma lista de objetos que estão a uma distância de 200 metros entre o ponto atual e o ponto do objeto localizado na árvore.

- 1: **Se** *raiz* contém um ou mais pontos
 - 2: **Então Se** *latitude* é igual a *raiz* -> *chave[0]* e *longitude* é igual a *raiz* -> *chave[1]*
Então listaRetorno -> *add(raiz->id)*.
Retorno
 - 3: Sendo *d* a distância entre o ponto A e o ponto B calculada utilizando a equação de Haversine em que a execução do algoritmo está descrita no Algoritmo 2.
 - 4: **Se** *d* for menor ou igual a 200
Então *listaRetorno* -> *add(raiz->id)*.
 - 5: *buscarProximos(raiz->esquerda, latitude, longitude, listaRetorno)*
 - 6: *buscarProximos(raiz->direita, latitude, longitude, listaRetorno)*
-

Na execução do Algoritmo 1, em uma de suas etapas, faz-se uso do cálculo da distância entre dois objetos utilizando a equação de Haversine (6). Essa equação leva em consideração a circunferência da terra para calcular a distância entre dois pontos a partir da latitude e a longitude desses dois pontos. Utilizada por navegadores antigos,

essa forma é facilitada pelos recursos computacionais atuais.

$$a = \sin^2\left(\frac{\Delta lat}{2}\right) + \cos(lat1) * \cos(lat2) * \sin^2\left(\frac{\Delta long}{2}\right)$$

$$c = 2 * a * \tan2(\sqrt{a}, \sqrt{1-a}) \quad (6)$$

$$d = R * c$$

Considera-se *R* como o maior raio da Terra estabelecido como 6.378,137 km da linha do equador para o DATUM WGS84. DATUM refere-se ao modelo matemático teórico da representação da superfície da Terra ao nível do mar utilizado pelos cartógrafos numa dada carta ou mapa. O Δlat é a diferença entre as duas latitude utilizadas, o $\Delta long$ é o mesmo padrão apenas referente à longitude (ASSIS, 2010). O Algoritmo 2 implementa o cálculo da equação de Haversine (6).

Algoritmo 2 – calcDistancia(*latA, longA, latB, longB, elA, elB*)

Entrada: Valores da latitude, longitude e altura do ponto A e do ponto B.

Saída: O valor da distância entre o ponto A e o ponto B.

- 1: Sendo *R* o valor do raio da terra
 - 2: Sendo *deltaLat* a diferença entre *latA* e *latB* multiplicada por *PI* e dividida por 180
 - 3: Sendo *deltaLong* a diferença entre *longA* e *longB* multiplicada por *PI* e dividida por 180
 - 4: Sendo *a* o seno ao quadrado de *deltaLat* somado com a multiplicação de cosseno de *latA* e cosseno de *latB* multiplicado por seno ao quadrado de *deltaLong*
 - 5: Sendo *c* o valor de 2 multiplicado por *atan2(raiz quadrada de a, raiz quadrada de 1-a)*
 - 6: Considere *d* a distância em metros sendo *R* multiplicado por *c * 1000*
 - 7: Sendo *alt* a diferença entre *elA* e *elB*
 - 8: Considere *d* sendo *d* elevado a 2 mais *alt* elevado a 2
 - 9: retorne raiz quadrada de *d*
-

A extração dos pontos chave considerados *inliers*, para a detecção e reconhecimento dos objetos em uma imagem de um local público ou turístico, descrito nessa metodologia, é realizada comparando os ângulos de orientação dos

gradientes locais e buscando a região onde se encontra a maior parte dos *inliers*. A Figura 12 mostra um exemplo de correspondência

entre os pontos chave providos pelo algoritmo SIFT.

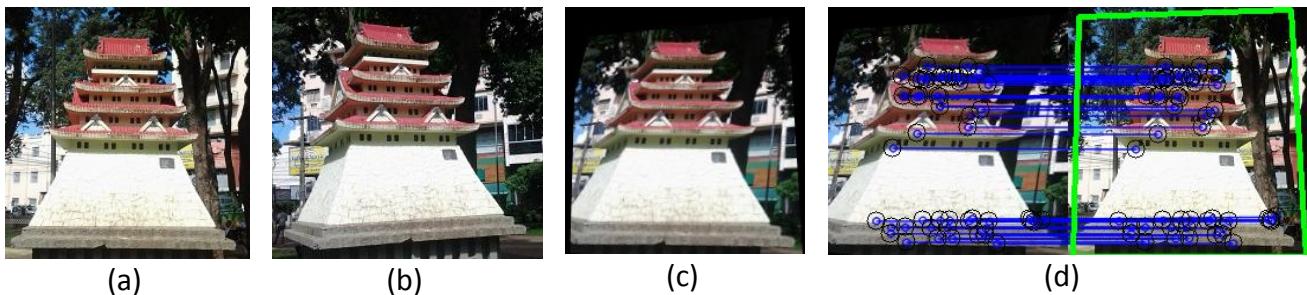


Figura 12. Correspondência entre os pontos chave após a execução do algoritmo SIFT. Em (a) tem-se a imagem do objeto cadastrado, em (b) a imagem enviada pelo aplicativo móvel, em (c) a imagem enviada pelo aplicativo móvel transformada para ser encaixar na imagem buscada e em (d) a imagem ilustrando a correspondência dos pontos chave entre a imagem do objeto cadastrado e a imagem enviada pelo aplicativo móvel.

A Figura 13 ilustra a tela do aplicativo móvel após a detecção de um objeto.

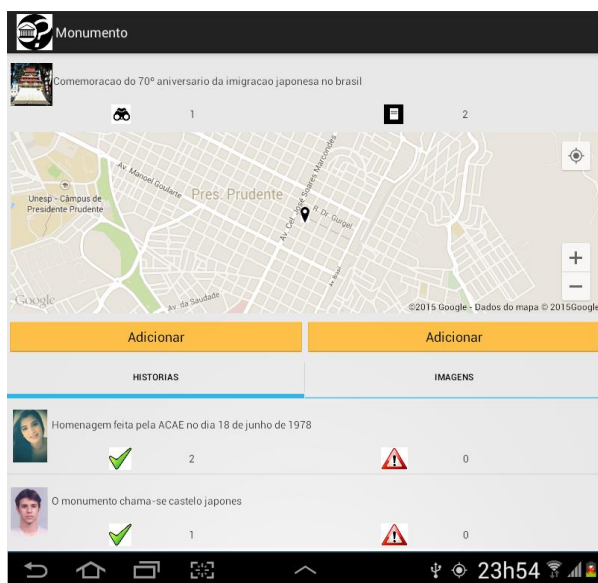


Figura 13. Tela do aplicativo móvel após a detecção e reconhecimento de um objeto.

Os conceitos da inteligência coletiva são aplicados neste trabalho por meio da troca de informações em pacotes XML, entre o aplicativo móvel e o *WebService*. Essa troca de informações ocorre em tempo real, distribuída a todos os usuários do Sistema colaborativo e o gerenciamento da

informação por meio das opções “curtir” e “denunciar” que permite aos usuários validar a veracidade da informação compartilhada e propor melhorias.

5 EXPERIMENTOS E RESULTADOS

Para realizar os experimentos foi utilizado, para instalação e execução do aplicativo móvel, um smartphone com processador de 1.4 GHz, memória de 1GB, câmera fotográfica de 8 MegaPixels e Sistema Operacional Android 4.0. O *WebService* foi instalado e configurado em um computador com processador Intel Core i3 3.30GHz, 6GB de memória RAM e Sistema Operacional Windows 8.1 64 bits.

Os experimentos foram realizados com 9 pares de imagens de objetos de locais públicos e turísticos. Dentre os 9 objetos cadastrados 7 estão localizados em um raio de 200 metros, desta forma foi possível

realizar testes não só no algoritmo de detecção como também no algoritmo de busca proposto (Algoritmo 2).

A Figura 14 apresenta a relação das imagens dos objetos cadastrados e as

imagens utilizadas para a busca e detecção do objeto. Nas imagens é possível observar os diferentes tamanhos e ângulos para cada objeto.

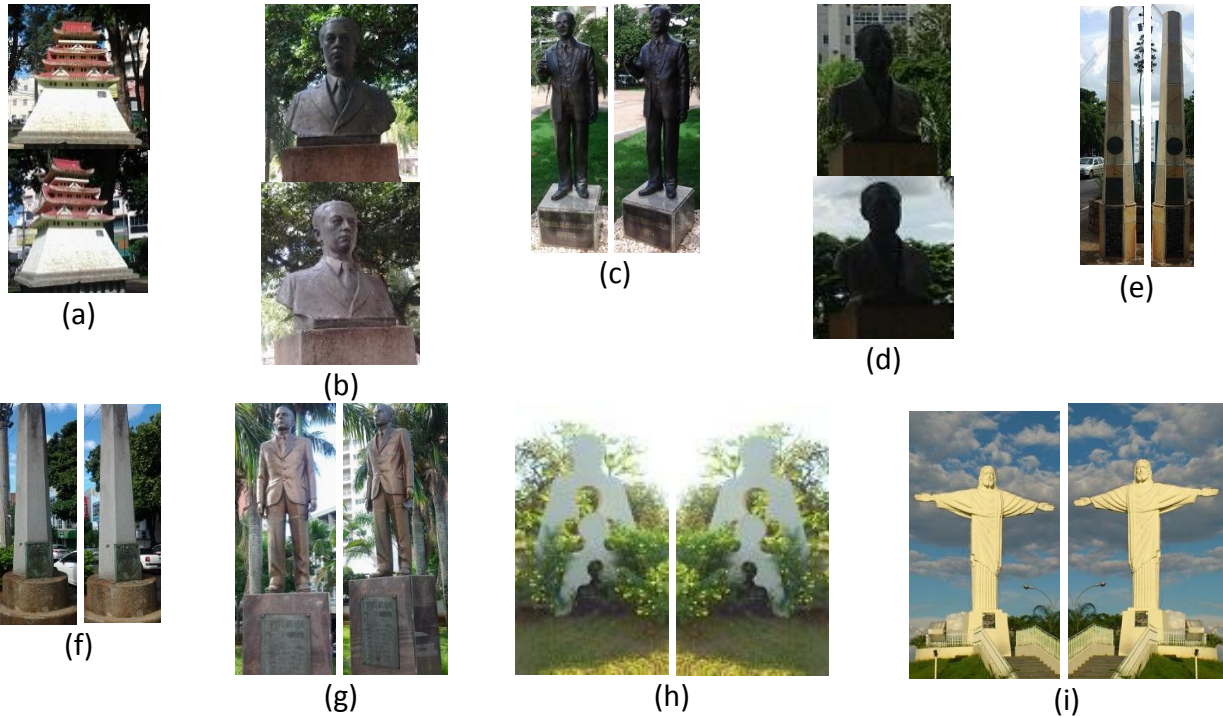


Figura 14. Conjunto de par de imagens de objetos em imagens de local público e turístico. (a) Castelo Japonês, (b) Dr. Jose A. Gonçalves Foz, (c) Francisco de Paula Goulart, (d) Major Felício Tarabay, (e) Cinquentenário da Revolução Constitucionalista, (f) Heróicos Soldados da F.E.B (Força Expedicionária Brasileira), (g) Florivaldo Leal, (h) Homenagem a revolução de 1932, (i) Cristo Redentor.

A Tabela 1 apresenta os resultados obtidos aplicando a metodologia proposta neste trabalho. A primeira coluna se refere aos índices das imagens utilizadas no experimento (Figura 14); a segunda à quantidade de objetos ao redor do ponto de latitude e longitude enviado pelo aplicativo móvel na captura da imagem; a terceira coluna à detecção e reconhecimento, que obtiveram no mínimo 50 por cento de pontos chave considerado *inliers* dentro da margem de mais ou menos 5 graus de rotação; e a

quarta coluna informa se houve erro no algoritmo de detecção e reconhecimento.

Nos 9 pares de imagens de objetos usados neste experimento, conforme apresentado na Figura 14, quatro objetos foram reconhecidos corretamente, obtendo-se uma taxa de acerto de 44%. A busca por objetos cadastrados ao redor, obteve 100% de acerto e não houve nenhum erro no momento da execução do algoritmo de busca, detecção e reconhecimento. Dos nove objetos, cinco não foram detectados (56%).

Tabela 1. Resultados da busca e reconhecimento de objetos em imagens de locais públicos e turísticos.

Objetos	Metodologia deste trabalho		
	Busca	Detectou	Erro
(A) Castelo Japonês	7	S	N
(B) Dr. Jose A. Gonçalves Foz	7	N	N
(C) Francisco de Paula Goulart	7	N	N
(D) Major Felício Tarabay	7	N	N
(E) Cinquentenário da Revolução Constitucionalista	7	N	N
(F) Heroicos Soldados da F.E.B	7	S	N
(G) Florivaldo Leal	7	N	N
(H) Homenagem a revolução de 1932	1	S	N
(I) Cristo Redentor	1	S	N
Total	9	4	0

A não detecção desses objetos ocorreu devido as imagens enviadas para detecção terem uma variação de orientação e ângulo acima do definido na metodologia.

6 CONSIDERAÇÕES FINAIS

Com o desenvolvimento deste trabalho, pôde-se mostrar a importância que, a Computação Visual por meio dos algoritmos de detecção e reconhecimento de objetos em imagens, aliados à troca de informações, por meio de dispositivos móveis e o uso da inteligência coletiva, podem favorecer os consumidores de locais públicos e turísticos a obterem acesso às informações referente aos objetos que compõe esses locais.

Algumas modificações podem ser desenvolvidas para aprimorar este trabalho, como por exemplo, ajustar o algoritmo para detectar e reconhecer objetos em imagens que possuem orientações maiores que a

margem definida nesta metodologia, e assim aumentar a possibilidade de detecção dos objetos. Permitir que os objetos detectados e reconhecidos sejam compartilhados com as redes sociais atuais, assim as informações históricas e culturais de cada objeto podem colaborar com outros usuários.

REFERÊNCIAS

- ALMEIDA, L.L. **Construção automática de imagens de super-resolução a partir de mosaicos formados por sequências de imagens**. 2013. 180 f. Tese (Doutorado) – Escola de Engenharia da Universidade de São Paulo, São Carlos - SP. <http://dx.doi.org/10.11606/t.18.2013.tde-03012014-100834>
- ASSIS, P.U.M. **Sistema de rastreamento de veículos para empresas de transporte utilizando navegação por satélite**. 2010. Monografia (Graduação) – Centro Universitário de Brasília – UniCEUB, Brasília, 2010.
- BENTLEY, J. L. Multidimensional binary search trees used for associative searching. **Communications of the ACM**, v.18, n.9,

p.509-517, 1975.

<http://dx.doi.org/10.1145/361002.361007>

BOOCH, G.; RUMBAUGH, J. JACOBSON, I. **UML: guia do usuário**. 2. ed. Rio de Janeiro: Elsevier, 2006.

BRASIL. Ministério do Turismo. Departamento de Estudos e Pesquisa. **Estatística básica de turismo**. Brasília: DEPEs, 2013. Disponível em: <www.dadosefatos.turismo.gov.br/dadosefatos/estatisticas_indicadores/estatisticas_indicadores_turismo_mundial/>. Acesso em: agosto de 2014.

BUENO, M. A influência dos smartphones no processo de compra. **Revista E-commerce Brasil**, ano 3, ed. 16, p.44, 2013.

COLLET, A. et al. Object recognition and full pose registration from a single image for robotic manipulation. In: IEEE INTERNATIONAL CONFERENCE ON ROBOTICS AND AUTOMATION, ICRA'09. **Proceedings...** p.48–55, 2009. <http://dx.doi.org/10.1109/robot.2009.5152739>

CRUZ, A.J. et al. O uso da inteligência coletiva e o método simplex na busca de minimizar custo utilizando dispositivos móveis. **Colloquium Exactarum**, v.6, n.2, p.140-156, 2014. <http://dx.doi.org/10.5747/ce.2014.v06.n2.e081>

FISCHLER, M.A.; BOLLES, R.C. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. **Communications of the ACM**, v.24, p.381–395, 1981. <http://dx.doi.org/10.1145/358669.358692>

FURTADO, V. **Collective intelligence in law enforcement - The WikiCrimes system, Information Sciences, Special Issue on Collective Intelligence**, Georgia, v.180, p.4-17, 2010.

HIBERNATE. **Relational Persistence for Idiomatic Java**. 2014. Disponível em: <<http://docs.jboss.org/hibernate/orm/4.3/manual/en-US/html/>>. Acesso em: fev 2014.

HOWE, J. **O poder das multidões: porque a força da coletividade está remodelando o futuro dos negócios**. 2. ed. Rio de Janeiro: Elsevier, 2009.

KSOAP2. **Ksoap2-Android**. 2014. Disponível em: <<https://code.google.com/p/ksoap2-android/>>. Acesso em: ago 2014.

LIMA, R.D.; MOTA G.L.A.; PINTO, P.E.D. Interface Luana: uma aplicação gráfica para o ensino da árvore binária Kd-tree. In: WORKSHOP OF UNDERGRADUATE WORKS (WUW), SIBGRAPI 2012, XXV CONFERENCE ON GRAPHICS, PATTERNS AND IMAGES. **Proceedings...** Universidade do Estado do Rio de Janeiro, 2012. p.87-92.

LÉVY, P. **A inteligência coletiva: por uma antropologia do ciberespaço**. 5. ed. São Paulo: Loyola, 2007.

LOWE, D.G. Distinctive image features from scale-invariant keypoints. **International Journal of Computer Vision**, v. 60, n.2, p.91-110, 2004. <http://dx.doi.org/10.1023/B:VISI.0000029664.99615.94>

LOWE, D.G. Object recognition from local scale-invariant features. In: IEEE INTERNATIONAL CONFERENCE ON COMPUTER VISION. **Proceedings...** Corfu, Greece, p.1150-1157, 1999. <http://dx.doi.org/10.1109/iccv.1999.790410>

LUCIO, D. R. **Um aplicativo para dispositivos móveis voltados para usuários de transporte público**. 2011. 56 f. Monografia (Graduação) - Universidade Tecnológica Federal do Paraná, Campo Mourão – PR.

MAIA, J.G.R. **Detecção e reconhecimento de objetos usando descritores locais**. 2010. 124 f. Tese (Doutorado) Departamento de Computação, Centro de Ciências, Universidade Federal do Ceará, Fortaleza - CE.

OKABE, T.; SATO, Y. Object recognition based on photometric alignment using RANSAC. In: COMPUTER SOCIETY CONF. ON COMPUTER VISION AND PATTERN RECOGNITION, **Proceedings...** 2003. p.221-228. <http://dx.doi.org/10.1109/cvpr.2003.1211357>

OPENCV. **Open Computer Vision Library**. 2014. Disponível em: <<http://sourceforge.net/projects/opencvlibrary/>>. Acesso em: fev 2014.

RODRIGUES, L. et al. Detecção e reconhecimento de objetos em imagens utilizando algoritmos de extração de pontos chave. **Colloquium Exactarum**, v.6, n.2, p.86-103, 2014. <http://dx.doi.org/10.5747/ce.2014.v06.n2.e078>

SAMET, H. **Foundations of multidimensional and metric data structures**. Morgan Kaufmann, 2006.

SEBRAE. CNTur. **Pesquisa: perfil dos turistas e do segmento de oferta**. Brasília, 2012. Disponível em: <[http://bis.sebrae.com.br/GestorRepositorio/ARQUIVOS_CHRONUS/bds/bds.nsf/E4EE72775D0F632A83257A0C00757D41/\\$File/NT00047646.pdf](http://bis.sebrae.com.br/GestorRepositorio/ARQUIVOS_CHRONUS/bds/bds.nsf/E4EE72775D0F632A83257A0C00757D41/$File/NT00047646.pdf)>. Acesso em: agos 2013.

SILVA, F.A. **Georreferenciamento automático de placas de sinalização com imagens obtidas com um sistema móvel de mapeamento**. 2012. Tese (Doutorado) - Escola de Engenharia de São Carlos da Universidade de São Paulo, São Carlos - SP. <http://dx.doi.org/10.11606/t.18.2012.tde-31072012-115700>

SILVA, F.A. et al. Evaluation of keypoint detectors and descriptors. In: WORKSHOP DE VISÃO COMPUTACIONAL - WVC 2013, 9. Rio de Janeiro. **Anais...** 2013.

TECHTUDO. **O que é Smartphone e para que serve?** 28 dez. 2011. Disponível em: <<http://www.techtudo.com.br/artigos/noticia/2011/12/o-que-e-smartphone-e-para-queserve.html>>. Acesso em: fev 2014.

TREIBER, M. **An Introduction to object recognition: selected algorithms for a wide variety of application**. London: SpringerVerlag, 2010. 201p. <http://dx.doi.org/10.1007/978-1-84996-235-3>

WITKIN, A.P. Scale-space filtering. In: THE INTERNATIONAL JOINT CONFERENCE ON ARTIFICIAL INTELLIGENCE OF BIOLOGICAL SYSTEMS. **Proceedings...** v.1, n.1, p.1019-1022, 1983.