

---

**Universidade Federal de Ouro Preto  
Instituto de Ciências Exatas e Aplicadas  
Colegiado de Engenharia de  
Computação**

***RoadScan: Sistema  
Crowdsourcing para Identificação  
de Qualidade de Pavimentos  
Asfálticos***

**Lucas Cedro de Lima**

**TRABALHO DE  
CONCLUSÃO DE CURSO**

**ORIENTAÇÃO:**  
Prof. Vicente J. Peixoto de Amorim

**Agosto, 2016  
João Monlevade/MG**

---



**Lucas Cedro de Lima**

***RoadScan: Sistema Crowdsourcing* para  
Identificação de Qualidade de Pavimentos  
Asfálticos**

Orientador: Prof. Vicente J. Peixoto de Amorim

Coorientador: Prof. Filipe Nunes Ribeiro

Monografia apresentada ao curso de Engenharia de Computação do Departamento de Computação e Sistemas da Universidade Federal de Ouro Preto como requisito parcial para obtenção do grau de Bacharel em Engenharia de Computação

**Universidade Federal de Ouro Preto**

**João Monlevade**

**Agosto de 2016**

L732r

Lima, Lucas Cedro.

RoadScan [manuscrito]: sistema crowdsourcing para identificação de qualidade de pavimentos asfálticos / Lucas Cedro Lima. - 2016.

61f.: il.: color; graf; tabs.

Orientador: Prof. Me. Vicente José Peixoto Amorim.

Monografia (Graduação). Universidade Federal de Ouro Preto. Instituto de Ciências Exatas e Aplicadas. Departamento de Computação e Sistemas de Informação.

1. Android (Programa de computador). 2. Programa de computador (software). 3. Pavimentos de asfalto - Qualidade. I. Amorim, Vicente José Peixoto. II. Universidade Federal de Ouro Preto. III. Título.

CDU: 004.4



MINISTÉRIO DA EDUCAÇÃO  
Universidade Federal de Ouro Preto – UFOP  
Colegiado do Curso de Engenharia de Computação  
Campus João Monlevade



## Curso Engenharia de Computação

### FOLHA DE APROVAÇÃO DA BANCA EXAMINADORA

#### *RoadScan: Sistema Crowdsourcing para Identificação de Qualidade de Pavimentos Asfálticos*

Lucas Cedro de Lima

Monografia apresentada ao Departamento de Computação e Sistemas da Universidade Federal de Ouro Preto como requisito parcial da disciplina CEA496 – Trabalho de Conclusão de Curso II, do curso de Bacharelado em Engenharia de Computação, e aprovada pela Banca Examinadora abaixo assinada:

Prof. Me. Vicente José Peixoto de Amorim  
Mestre em Ciência da Computação pela Universidade Estadual de Campinas – SP, Brasil  
Orientador  
Departamento de Computação e Sistemas/DECSI – UFOP

Prof. Me. Filipe Nunes Ribeiro  
Mestre em Ciência da Computação pela Universidade Federal de São Carlos – SP, Brasil  
Coorientador  
Departamento de Computação e Sistemas/DECSI - UFOP

Prof. Me. Igor Muzetti Pereira  
Mestre em Ciência da Computação pela Universidade Federal de Ouro Preto – MG, Brasil  
Examinador  
Departamento de Computação e Sistemas/DECSI – UFOP

João Monlevade, 12 de agosto de 2016



## ATA DE DEFESA

Aos 12 dias do mês de agosto de 2016, às dez horas, no Laboratório de Banco de Dados e Redes (C304), foi realizada a defesa de Monografia pelo aluno Lucas Cedro de Lima, sendo a Comissão Examinadora constituída pelos professores: Prof. Me. Vicente José Peixoto de Amorim, Prof. Me. Filipe Nunes Ribeiro e Prof. Me. Igor Muzetti Pereira. O candidato apresentou a monografia intitulada: “*RoadScan: Sistema Crowdsourcing para Identificação de Qualidade de Pavimentos*”. A comissão examinadora deliberou, por unanimidade, pela aprovação do candidato, concedendo-lhe o prazo de 15 dias para incorporação no texto final das alterações sugeridas. Na forma regulamentar, foi lavrada a presente ata que é assinada pelos membros da Comissão Examinadora e pelo formando.

João Monlevade, 12 de agosto de 2016.

Prof. Me. Vicente José Peixoto de Amorim  
Professor Orientador/Presidente

Prof. Me. Filipe Nunes Ribeiro  
Professor Coorientador

Prof. Me. Igor Muzetti Pereira  
Professor Convidado

Lucas Cedro de Lima  
Lucas Cedro de Lima



UNIVERSIDADE FEDERAL DE OURO PRETO  
INSTITUTO DE CIÊNCIAS EXATAS E APLICADAS  
COLEGIADO DE ENGENHARIA DE COMPUTAÇÃO

---

## ANEXO II – TERMO DE RESPONSABILIDADE

### Curso Engenharia de Computação

#### TERMO DE RESPONSABILIDADE

Eu, Lucas Pedro de Lima,  
declaro que o texto do trabalho de conclusão de curso intitulado  
“Road Scan Sistema Crowdsourcing para Identificação de Qualidade de Pavimentos Asfálticos” é de  
minha inteira responsabilidade e que não há utilização de texto, material fotográfico, código  
fonte de programa ou qualquer outro material pertencente a terceiros sem as devidas  
referências ou consentimento dos respectivos autores.

João Monlevade, 31 de Agosto de 2016

Lucas Lima

Assinatura do aluno

*Dedicado à minha família e amigos.*

# Agradecimentos

Agradeço primeiramente aos meus pais, Robinson e Márcia, que acreditaram em mim no começo desta jornada e me apoiam desde então. Aos meus irmãos, Bruno e Leandro, pelas experiências e companheirismo.

Agradeço aos meus amigos e companheiros de estudo, Leandro, Luíza e Welton pela nossa dedicação e amizade que me sustentaram nessa vitória.

Agradeço ao meu orientador, Prof. Vicente J. Peixoto Amorim, pelo incentivo e paciência. Ao Prof. Igor Muzetti pela gerência, que possibilitou a execução de um projeto de qualidade. E ao laboratório iMobilis, pelas amizades e pelo amadurecimento acadêmico e profissional.

Por fim agradeço à minha segunda família destes últimos anos, irmãos da república Diretoria. Obrigado a todos pela amizade, apoio, respeito e ótima convivência. Os momentos sóbrios estarão sempre em minha memória.



*“O mais competente não discute, domina a sua ciência e cala-se.”*

*Voltaire*



# Resumo

Rodovias e estradas danificadas são motivos de desconforto, prejuízos e fatalidades à comunidade. Além disso, devido ao sua difícil e dispendiosa fiscalização, este problema é muitas vezes negligenciado. Estudos já foram realizados para a resolução deste problema utilizando dispositivos móveis, como os *smartphones*, de forma simples e menos custosa, permitindo determinar a qualidade de pistas asfálticas. Através de tais dispositivos é possível apontar áreas de atenção redobrada aos motoristas e às autoridades responsáveis pela manutenção. As propostas existentes tendem a utilizar o padrão internacional IRI (*International Roughness Index*) que demanda muito processamento, sendo geralmente necessário o uso de equipamentos e *softwares* adicionais. O trabalho aqui apresentado (“*RoadScan*”) é uma aplicação Android desenvolvida com o propósito de determinar a qualidade de pisos asfálticos através da análise dos dados coletados pelo acelerômetro e GPS em posições arbitrárias. Os resultados das coletas de diversos usuários são exibidos em uma página pública utilizando o Google Maps, com cores intuitivas relacionadas a qualidade. Os resultados são promissores identificando na maioria dos casos, as regiões que apresentam algum tipo de deformidade.

**Palavras-chaves:** Desvio Padrão, Crowdsourcing, Android, Pavimento, IRI, Acelerômetro.



# Abstract

Damaged highways and roads are cause for discomfort, money prejudice and casualties to the community. In addition to that, due to its expensive and complicated oversight, this problem is being commonly neglected. Studies have already been carried out for the resolution of this issue using mobile devices such as *smartphones*, in a simpler and less costly way. Through such devices it is possible to determine the quality of asphalt tracks and point out areas of attention to drivers and the authorities responsible for the maintenance. Existing proposals tend to use the international standard IRI (*International Roughness Index*) that demand a lot of processing, being generally required the use of additional equipment and *softwares*. The work presented here ("*RoadScan*") is an Android application developed with the purpose of determining the quality of asphalt floors through the analysis of data collected by the accelerometer and GPS in arbitrary positions. The results of the collections of several users are displayed on a public page using Google Maps, with intuitive colors related to quality. The results are promising in most cases, identifying the regions that present some kind of deformity.

**Key-words:** Standard Deviation, Crowdsourcing, Android, Pavement, IRI, Acelerometer.



# Lista de ilustrações

|  |    |
|--|----|
| Figura 1 – Modelo em camadas do Android . . . . .                                    | 27 |
| Figura 2 – Sistema de coordenadas . . . . .  | 28 |
| Figura 3 – Android Studio IDE . . . . .  | 40 |
| Figura 4 – Leituras do acelerômetro e desvio padrão de uma pista real . . . . .      | 41 |
| Figura 5 – Fotos da tela do <i>smartphone</i> . . . . .                              | 42 |
| Figura 6 – Fluxo de Coleta dos Dados . . . . .                                       | 43 |
| Figura 7 – Elementos armazenados dentro de cada segmento de pista . . . . .          | 44 |
| Figura 8 – Exemplo do algoritmo de detecção . . . . .                                | 46 |
| Figura 9 – Fluxo de tratamento dos dados . . . . .                                   | 46 |
| Figura 10 – Categorização das quatro qualidades de pavimentos . . . . .              | 47 |
| Figura 11 – Retângulos de busca . . . . .  | 48 |
| Figura 12 – Mapa resultante referenciando as qualidades e os trechos reais . . . . . | 53 |



# Lista de tabelas

|   |    |
|---|----|
| Tabela 1 – Flags e cores correspondentes. . . . . | 42 |
|---|----|



# Lista de abreviaturas e siglas

|      |  |
|------|--|
| CNT  | Confederação Nacional do Transporte    |
| IRI  | International Roughness Index          |
| GPS  | Sistema de Posicionamento Global       |
| API  | Interface de Programação de Aplicações |
| HTML | Linguagem de Marcação de Hipertexto    |
| JSON | Notação de Objetos JavaScript          |
| AJAX | Javascript e XML Assíncrono            |



# Sumário

|                |   |           |
|----------------|---|-----------|
|                | <b>Introdução</b> . . . . .   | <b>23</b> |
| <b>I</b>       | <b>REFERENCIAIS TEÓRICOS</b>  | <b>25</b> |
| <b>1</b>       | <b>REVISÃO BIBLIOGRÁFICA</b> . . . . .                              | <b>27</b> |
| <b>1.1</b>     | <b>Android</b> . . . . .  | <b>27</b> |
| <b>1.2</b>     | <b>Acelerômetro</b> . . . . .                                       | <b>28</b> |
| <b>1.3</b>     | <b>CrowdSourcing</b> . . . . .                                      | <b>28</b> |
| <b>2</b>       | <b>PESQUISAS RELACIONADAS</b> . . . . .                             | <b>31</b> |
| <b>II</b>      | <b>PREPARAÇÃO DA PESQUISA</b>                                       | <b>33</b> |
| <b>3</b>       | <b>MOTIVAÇÃO E OBJETIVOS</b> . . . . .                              | <b>35</b> |
| <b>3.1</b>     | <b>Motivação</b> . . . . .  | <b>35</b> |
| <b>3.2</b>     | <b>Objetivos</b> . . . . .  | <b>36</b> |
| <b>3.2.1</b>   | <b>Aplicativo</b> . . . . .   | <b>36</b> |
| <b>3.2.1.1</b> | Requisitos Funcionais . . . . .                                     | 36        |
| <b>3.2.1.2</b> | Requisitos Não-Funcionais . . . . .                                 | 36        |
| <b>3.2.2</b>   | <b>Servidor</b> . . . . .   | <b>36</b> |
| <b>3.2.2.1</b> | Requisitos Funcionais . . . . .                                     | 36        |
| <b>3.2.2.2</b> | Requisitos Não-Funcionais . . . . .                                 | 37        |
| <b>4</b>       | <b>MATERIAIS E MÉTODOS</b> . . . . .                                | <b>39</b> |
| <b>4.1</b>     | <b>Equipamentos utilizados</b> . . . . .                            | <b>39</b> |
| <b>4.1.1</b>   | <i>Smartphones</i> Android . . . . .                                | 39        |
| <b>4.1.2</b>   | Veículos . . . . .  | 39        |
| <b>4.1.3</b>   | Máquina Servidora . . . . .   | 39        |
| <b>4.2</b>     | <b>Softwares utilizados</b> . . . . .                               | <b>40</b> |
| <b>4.2.1</b>   | Android Studio . . . . .  | 40        |
| <b>4.2.2</b>   | Linguagens de Programação . . . . .                                 | 40        |
| <b>4.3</b>     | <b>Metodologia de desenvolvimento</b> . . . . .                     | <b>41</b> |
| <b>4.3.1</b>   | Coleta de Dados . . . . .   | 43        |
| <b>4.3.2</b>   | Processamento . . . . .   | 43        |
| <b>4.3.2.1</b> | Casos Extremos - Identificação de Trechos Bons e Péssimos . . . . . | 45        |

|            |  |           |
|------------|--|-----------|
| 4.3.2.2    | Casos Intermediários - Identificação de Trechos Médios e Ruins . . . . . | 45        |
| 4.3.2.3    | Caso Isolado - Identificação de Problemas Pontuais . . . . .             | 47        |
| 4.3.3      | Comunicação Cliente-Servidor . . . . .                                   | 47        |
| 4.3.4      | Servidor . . . . .   | 47        |
| 4.3.4.1    | Leitura dos Dados e Desenho das Linhas . . . . .                         | 48        |
| 4.3.4.2    | Tratamento de Problemas Pontuais . . . . .                               | 49        |
| 4.3.4.3    | Condensamento das Informações . . . . .                                  | 49        |
| <b>III</b> | <b>RESULTADOS</b>  | <b>51</b> |
|            | <b>Conclusão . . . . .</b>   | <b>55</b> |
| <b>5</b>   | <b>TRABALHOS FUTUROS . . . . .</b>                                       | <b>57</b> |
|            | <b>REFERÊNCIAS . . . . .</b>   | <b>59</b> |

# Introdução

A qualidade das pistas/pisos asfálticos das estradas brasileiras afeta diariamente a vida de muitos motoristas, maximizando a chance de acidentes e minimizando o conforto. Devido a extensa malha rodoviária e falta de métodos práticos de baixo custo para avaliar sua qualidade, este tema é frequentemente negligenciado pelos órgãos responsáveis. De acordo com o relatório gerencial (GERENCIAL, 2015) da Confederação Nacional do Transporte (CNT), 57,3% das rodovias de gestão pública avaliadas apresentam alguma deficiência, sendo 24,3% abaixo de regular. Para rodovias de gerência federal estes números atingem uma margem parecida de 51% com deficiência, sendo 15,1% abaixo de regular.

A técnica empregada para a classificação da qualidade de pavimentos difere-se em cada região do globo (HALL; LEWIS; WOLFE, 2013), sendo estas de caráter manual ou automático. O primeiro é realizado através de análises visuais de trechos ou por equipamentos que realizam a leitura de pequenos segmentos de pista por vez. Metodologias manuais, quando comparadas as automáticas, são mais lentas e imprecisas, além de necessitar mais indivíduos para sua execução. Métodos automáticos usualmente consistem de veículos com câmeras e lasers acoplados ao mesmo, sendo alguns capazes de trafegar em velocidades de aproximadamente 80km/h, dependendo de sua sofisticação. De qualquer maneira, ambas tecnologias requerem alto custo financeiro para os equipamentos e equipe treinada para o manuseio (MELOROSE; PERROY; CAREAS, 2015a)

Sabe-se que a forma e a qualidade de pavimentos asfálticos são alteradas com o tempo devido a fatores externos (ADLINGE; GUPTA, 2009), sendo necessária uma constante atualização de tais informações. Porém, todas as alternativas são dispendiosas em tempo e dinheiro devido a lentidão no processo de recuperação dos dados sobre as pistas e/ou demora na aquisição de equipamentos onerosos, entre outros fatores, como por exemplo o bloqueio por completo das pistas para análise. Também é válido mencionar que a qualidade do piso asfáltico pode afetar negativamente no consumo de combustível do automóvel (KLAUBERT, 2001).

Devido a impraticabilidade destes métodos, diversos estudos (CHUGH; BANSAL; SOFAT, 2014) procuram resolver este desafio de maneira menos dispendiosa. A alternativa mais promissora atualmente tem sido os *smartphones*, dispositivos móveis de alta popularidade que possuem sensores embutidos capazes de coletar informações precisas e relevantes referentes a este contexto. O sensor de maior destaque é o acelerômetro, que é capaz de identificar as vibrações e o impacto aplicado no aparelho. Quando em contato com o veículo o sensor captura indiretamente as vibrações do mesmo, tornando possível a identificação de falhas e irregularidades presentes na pista.

A maioria dos projetos na área procuram traduzir a saída do sistema para o índice internacional IRI (*International Roughness Index*) (SAYERS; GILLESPIE; PATERSON, 1986)(PFLEEGER, 2006). Apesar de ser uma métrica mundialmente reconhecida, esta requer muito entendimento e processamento matemático complexo. Estas operações exigem muito da capacidade de processamento do *smartphone*, sendo geralmente realizadas por aparelhos e *softwares* específicos. Por fim, assim como nosso projeto, algumas destas soluções ((DOUANGPHACHANH; ONEYAMA, 2013)(CHEN et al., 2011)(ARHIN et al., 2015)) realizam a tradução de intervalos de IRI com qualidade asfáltica, porém utilizamos aqui uma nova abordagem para a classificação.

No atual cenário de sistemas e aplicativos capazes de classificar pavimentos, são encontrados desafios como: a posição do aparelho, processamento e atualização dos dados. Os dispositivos usualmente devem ser fixados em alguma superfície plana para uma leitura precisa do eixo vertical do acelerômetro, ferindo a usabilidade. Alguns sistemas utilizam de *softwares* externos (DOUANGPHACHANH; ONEYAMA, 2014) para o processamento dos dados, necessitando equipamentos dedicados ao serviço. A maioria das propostas visa a classificação da qualidade de pavimentos, mas não uma maneira de espalhar o sistema em grande escala de forma integrada.

Na intenção de contribuir com a solução desta problemática, é apresentado aqui a aplicação *RoadScan*. Um sistema que visa mapear a qualidade de pistas utilizado *smartphones* em posições arbitrárias, tendo como maior diferencial a análise de desvios padrões e seus antecessores para determinar a qualidade dos trechos, separando-os por faixas de velocidade. Todo o processamento necessário é realizado no próprio aparelho, demonstrando em tempo real a qualidade do pavimento traduzido em cores intuitivas. As informações coletadas são armazenadas em um servidor dedicado onde é feita a leitura dos dados para a execução do algoritmo *crowdsourcing*. A exibição dos resultados é realizada utilizando-se a API GoogleMaps, sendo estes atualizados conforme a obtenção de novas contribuições.

Parte I

Referenciais Teóricos



# 1 Revisão Bibliográfica

## 1.1 Android

Android ([ANDROID, 2016](#)) é o nome dado ao sistema operacional hoje gerenciado pela Google, que opera principalmente em *tablets* e *smartphones*. A tecnologia está presente no mercado há mais de 8 anos e a empresa disponibiliza constantes atualizações ao sistema oferecendo mais recursos e praticidade aos usuários.

Baseado em kernel Linux de código aberto, o *framework* oferece APIs que permite o controle dos vários módulos presentes no aparelho, como GPS, sensores de aceleração, luminosidade, entre outros. Por ser de código aberto, este oferece a possibilidade de ser modificado pelos usuários para que possam personalizar suas próprias versões Android ([HILDENBRAND, 2015](#)). Esta é uma prática comumente utilizada pelas empresas que revendem os aparelhos, alterando sua interface e criando uma identidade referente à marca.

Em meados de 2015 houve um crescimento de 13% no número de vendas de aparelhos Android, chegando a uma fatia de 82,8% do mercado no número de vendas em relação aos seus concorrentes([IDC, 2015](#)). Tais valores representam o potencial que a tecnologia carrega e o quão promissor é criar soluções para este ambiente móvel.

O sistema possui uma divisão em camadas como mostra a figura 1. Esta separação permite a compatibilidade e portabilidade para outros aparelhos, como por exemplo os dispositivos vestíveis *wearables* ([SUNG, 2015](#)), que possuem diversas funcionalidades, como os relógios inteligentes (*smartwatches*).

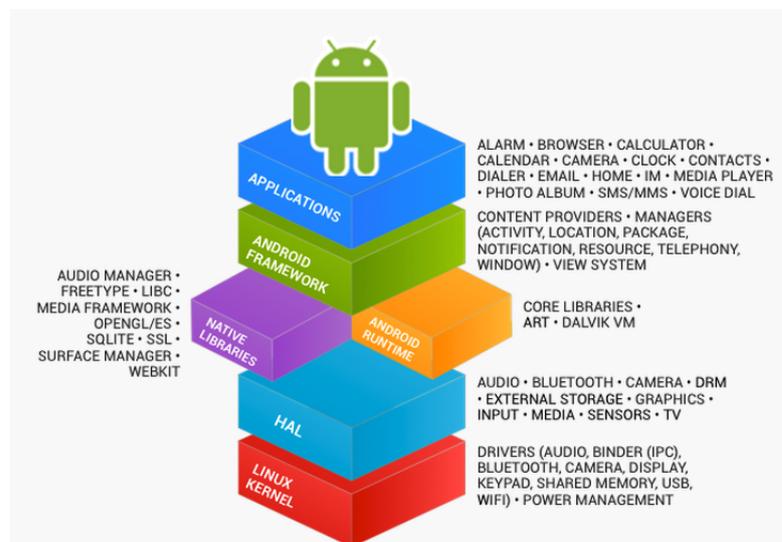


Figura 1: Modelo em camadas do Android ([ANDROID. . . , 2016](#))

O Android foi escolhido para ser usado neste trabalho devido ao seu baixo custo e alta popularidade. Estes fatores são interessantes pois o projeto apresenta o caráter de conhecimento coletivo, necessitando o maior número possível de usuários.

## 1.2 Acelerômetro

Muitos dos aparelhos Android possuem embutidos diversos sensores (DEVELOPERS, 2016) capazes de medir movimento, orientação e condições do ambiente. Com a tecnologia atual os mesmos são capazes de fornecer dados de alta precisão que são utilizados para diversos fins em diferentes aplicações.

Para a realização deste projeto foi utilizado principalmente o sensor de movimento conhecido como acelerômetro. Tal sensor mede a força de aceleração em  $m/s^2$  aplicado no dispositivo nos 3 eixos x,y e z como mostra a figura 2. Quando em repouso, o aparelho irá obter a leitura de aproximadamente  $9,8 m/s^2$ , correspondente a força gravitacional da terra.

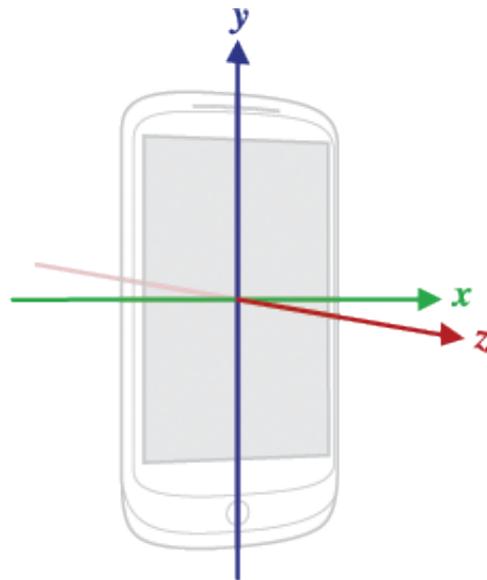


Figura 2: Sistema de coordenadas (DEVELOPERS, 2016)

Com a discriminação dos eixos, é possível identificar a direção da força aplicada no aparelho. Esta será a ferramenta principal do trabalho já que é desejado identificar o movimento vertical do veículo e a força do impacto nele aplicada.

## 1.3 CrowdSourcing

*CrowdSourcing* (IDEA, 2016) é o termo utilizado para a divisão de um trabalho em pequenas partes, sendo usualmente terceirizado para colaboradores de maneira *online*.

O trabalho pode ser realizado por voluntários ou por trabalhadores e algumas plataformas podem oferecer algum tipo de recompensa pela tarefa realizada.

Esta ideia é implementada de diversas maneiras, como o *crowdfunding* que busca interessados dispostos a doar dinheiro para a realização de projetos (ex.: *Kickstarter* (KICKSTARTER, 2016)); Micro-tarefas onde os usuários devem realizar alguma pequena atividade em troca de pequenas recompensas monetárias, como o aplicativo WHAFF (WHAFF, 2016); Conhecimento coletivo, em que usuários contribuem para um conjunto ou repositório de conhecimento (Wikipédia (WIKIPEDIA, 2016)).

Por sua vez, o *Mobile Crowdsourcing* é o ramo voltado para dispositivos móveis, que graças aos poderosos recursos como GPS, câmeras e sensores, são capazes de fornecer e construir informações com alto grau de confiança. Os dados podem ser coletados por tais dispositivos de forma passiva, em segundo plano, ou de forma ativa como por exemplo: através do envio de fotos relevantes, avaliação de prestação de serviços, participação em pesquisas através de preenchimento de formulários como o *Google Opinion Rewards* (GOOGLE..., 2016), entre outros.

De um lado, o *crowdsourcing* é uma técnica que terceiriza o esforço de maneira barata, rápida e com alta diversidade em seu conjunto amostral. Porém, em contrapartida, o incentivo a continuidade da contribuição dos usuários e a fidelidade das informações geradas é uma adversidade que deve ser considerada.

O trabalho aqui apresentado *RoadScan* usufrui desta ideia para manter a atualização dos dados, definir a fidelidade da informação e para a remoção de possíveis *outliers*, valores atípicos que são inconsistentes com os demais considerando-se a análise de uma região específica.



## 2 Pesquisas Relacionadas

Muitos sistemas inteligentes para monitorar qualidade de pavimentos estão sendo desenvolvidos ao redor do globo. Estes diferem entre si principalmente no tipo de *hardware* utilizado para captação dos dados, posição dos sensores, processamento local ou em nuvem e público-alvo para o qual serão disponibilizados os resultados, como pode ser visto em (CHUGH; BANSAL; SOFAT, 2014). Nota-se aumento do número de trabalhos publicados nesta área no decorrer dos últimos anos. Isto se deve ao fato de que os resultados estão sendo satisfatórios utilizando equipamentos mais acessíveis, no caso, os *smartphones*.

Abaixo é apresentada uma listagem dos trabalhos que apresentam similaridade com o RoadScan:

1. **Roadroid**: Aplicação Android desenvolvida por uma equipe sueca que trabalha no ramo desde os anos 90 (MELOROSE; PERROY; CAREAS, 2015b). Esta aplicação utiliza acelerômetros e GPS e está disponível na plataforma online (ROADROID, 2016) oficial dos desenvolvedores. Utilizam o IRI como base para calcular seus próprios índices denominados: eIRI e cIRI. Obtiveram uma taxa de precisão de 81% em relação aos valores coletados via laser. RoadDroid também é o único que disponibiliza os dados de forma pública e utiliza o conceito *crowdsourcing*.
2. **StreetBump** (OFFICE, 2016): Aplicação para iOS que atualmente monitora qualidade de pavimentos através da detecção de buracos e quebra-molas das ruas de Boston (EUA). A grande problemática desta abordagem é que os dados são fornecidos apenas e diretamente aos órgãos responsáveis, não provendo à comunidade o retorno de suas avaliações, apesar de ser esta uma funcionalidade prevista para o futuro. Esta aplicação pode ser encontrada gratuitamente na Apple Store.
3. **Nericell** (MOHAN; RAMJEE; PADMANABHAN, 2008): Projeto realizado pelo grupo Microsoft Research. Além da qualidade de pavimentos, monitora também a qualidade do tráfego em uma dada região, utilizando algoritmos baseado na ativação e leitura do microfone do *smartphone*. Apresenta a possibilidade de utilizar o dispositivo em qualquer posição realizando a virtualização dos eixos e captação da frenagem do automóvel para identificar a direção da locomoção. Este trabalho é uma continuação do projeto TrafficSense (MOHAN; PADMANABHAN; RAMJEE, 2008) desenvolvido pelos mesmos.
4. **Pothole Patrol** (ERIKSSON; GIROD; HULL, 2008): Também conhecido como P<sup>2</sup>, esse trabalho serviu como base para diversos outros, como o item 3 e , e o próprio RoadScan. Sua contribuição provém do algoritmo “*z-peak*” de detecção de buracos

desenvolvido pelos autores, que utiliza limiares, baseados na leitura, que quando ultrapassados identificam buracos na pista. .

5. **Real Time Pothole Detection Using Android Smartphones With Accelerometers** (MEDNIS et al., 2011): Este estudo apresenta uma melhora do algoritmo utilizado no item 4. Além do “*Z-THRESH*” que é similar ao “*z-peak*”, é descrito a utilização de mais três abordagens para ajudar na confirmação de presença de buracos: o “*Z-DIFF*” e “*STDEV(Z)*” que procuram por uma diferença grande o suficiente entre dois valores consecutivos, da leitura do acelerômetro e de seu desvio padrão respectivamente; e “*G-ZERO*” que após um pico identificado, verifica se os três eixos atingem valores próximos de 0, que foi um comportamento comum notado pelos pesquisadores.

RoadScan apresenta similaridade com o algoritmo apresentado em 4 e 5, identificando picos consecutivos nos desvios padrões em um pequeno intervalo para a identificação não apenas de buracos, mas de segmentos de pista que apresentem deformidades. Além disso, o projeto também contempla a ideia de “*crowdsourcing*” visto em (ROADROID, 2016). Visando a conscientização da população sobre os problemas existentes, esta metodologia é implantada para manter os dados sempre atualizados. Todos os resultados podem ser consultados de forma transparente através da página online.

## Parte II

### Preparação da pesquisa



## 3 Motivação e Objetivos

### 3.1 Motivação

Rodovias apresentam alta importância para segurança e economia, como visto em (BARTHOLOMEU; Caixeta Filho, 2008). Devido a falta de técnicas e equipamentos de baixo custo para o mapeamento de estradas, surge a necessidade da procura por métodos mais viáveis e acessíveis. Milhares de quilômetros devem ser percorridos lentamente e re-avaliados para se mapear as principais rodovias de um país. A falta de praticidade e alto custo financeiro são características expressivas nas metodologias atuais, podendo ser muitas vezes descartadas ou realizadas de maneira insuficiente.

Sabe-se também que o estado de qualidade de rodovias muda constantemente (ADLINGE; GUPTA, 2009). Chuvas, calor, neve e o constante tráfego de veículos pesados resultam em buracos e falhas que podem danificar o veículo, gerar desconforto e acidentes. Torna-se então necessário um sistema que além de mapear, seja capaz de atualizar constantemente o estado corrente das pistas, para que o motorista esteja sempre a par dos possíveis perigos que a pista lhe apresenta.

A implantação de uma metodologia *crowdsourcing* torna-se promissora já que o número de usuários de *smartphones* cresce a cada dia (IDC, 2015), facilitando sua escalabilidade. Com a criação de uma aplicação simples e intuitiva que requer o menor contato possível com o usuário, é possível a realização de um trabalho complexo e dispendioso de uma maneira fácil e eficiente. A utilização do conhecimento coletivo contorna também o problema de países com malha rodoviária muito extensa, como o Brasil, já que o trabalho será sempre dividido em múltiplos contribuintes.

Mesmo havendo a inspeção destas vias pelos órgãos responsáveis, os mesmos disponibilizam à população apenas números, mas não a localização dos problemas. É de grande valia disponibilizar à comunidade os resultados de suas coletas de forma a identificar o local exato do problema, como forma de aumentar a atenção dos moradores e incentivar a demanda por providências.

Pode se dizer com alto grau de confiança que não existe nenhum trabalho brasileiro publicado neste ramo. Apenas um trabalho que procura validar o uso de rede neurais para relacionar a vibração do chassi e a qualidade da pista (FALLIS, 2013). Porém não visa a criação de um sistema público para dispositivos móveis com interface *web* acessível voltado aos usuários.

## 3.2 Objetivos

O objetivo principal deste trabalho foi a criação de um sistema composto por: uma aplicação Android, capaz de determinar de maneira mais simples a qualidade de pavimentos em tempo real; e uma interface *web* para acesso aos resultado contando com o conhecimento coletivo para redução dos erros.

Abaixo são descritos os requisitos do sistema separados em duas classificações, funcionais e não funcionais. Requisitos funcionais definem a maneira que o sistema deve se comportar a determinadas entradas ou situações, ou seja, como ele deve funcionar. Em contrapartida, requisitos não-funcionais especificam propriedades e restrições do sistema, ou seja, como ele deve ser.

### 3.2.1 Aplicativo

#### 3.2.1.1 Requisitos Funcionais

1. A aplicação deve utilizar os sensores de acelerômetro e GPS para coleta de dados;
2. A coleta deve ser iniciada apenas após conexão com o sistema de localização;
3. A aplicação deve operar em posições arbitrárias;
4. Após processamento, os dados devem ser armazenados em banco de dados local;
5. Os dados devem ser enviados ao servidor quando conexão com a internet for identificada;
6. Os dados devem ser apagados do banco após confirmação de envio; e

#### 3.2.1.2 Requisitos Não-Funcionais

1. A aplicação deve ser capaz de realizar todo o processamento localmente em tempo real;
2. Os dados devem ser enviados utilizando formatação JSON.
3. A aplicação deve requerer o mínimo de contato e conhecimento do usuário.

### 3.2.2 Servidor

#### 3.2.2.1 Requisitos Funcionais

1. Deve hospedar a página responsável pela mostragem do mapa;
2. Deve hospedar o banco de dados responsável pelo armazenamento dos dados;

3. Deve ser carregado apenas os pontos contidos na porção do mapa sendo exibida;
4. Deve executar rotina de condensamento de dados
5. As linhas devem ser desenhadas com cores referentes as suas qualidades;
6. Cada linha deve conter uma janela de informações sobre o trecho.

#### 3.2.2.2 Requisitos Não-Funcionais

1. A página deve carregar rápido suficiente para dar ao usuário noção de interatividade.



## 4 Materiais e métodos

Para a execução deste projeto foi necessário a realização de testes utilizando diferentes *smartphones* em conjunto com alguns modelos de automóveis. Aqui serão apresentados os modelos dos aparelhos e veículos utilizados assim como os testes necessários para a comprovação de algumas teorias fundamentais para o desenvolvimento do projeto.

### 4.1 Equipamentos utilizados

#### 4.1.1 *Smartphones* Android

Como apontado na seção de objetivos, o projeto nesta etapa cobre apenas o sistema operacional Android, devido sua popularidade em nossa região e por disponibilidade de recursos. A base atual de dados é composta pelas informações coletadas dos seguintes aparelhos:

- Xiaomi - Redmi 2 Pro
- Motorola - Moto G 1ª Geração
- Motorola - Moto X 1ª Geração
- Motorola - Moto X 2ª Geração
- Asus - ZenFone

#### 4.1.2 Veículos

Os testes foram realizados nos seguintes automóveis:

- Citroën - C3
- Volkswagen - Gol
- Fiat - Uno
- Honda - Civic
- Toyota - Corolla

#### 4.1.3 Máquina Servidora

Toda a interface *web* assim como o *web service* foram implementados em uma máquina servidora HP Compaq Elite 8300 SFF com um processador Intel(R) Core(TM) i5-3570 fde 3.40GHz e 8GB de memória RAM DDR3 rodando o Ubuntu Server 12.04LTS de 64bits.

## 4.2 Softwares utilizados

### 4.2.1 Android Studio

Android Studio (Figura 3) é a IDE oficial para desenvolvimento em Android disponibilizado pela Google e baseado no IntelliJ IDEA.

Essa plataforma oferece diversos recursos para o desenvolvimento, como a depuração de código, emulador virtual nativo e ferramentas para realização de testes, integração com o sistema de gerenciamento de código GitHub, monitor de recursos, além de oferecer uma interface agradável e contar com frequentes atualizações.

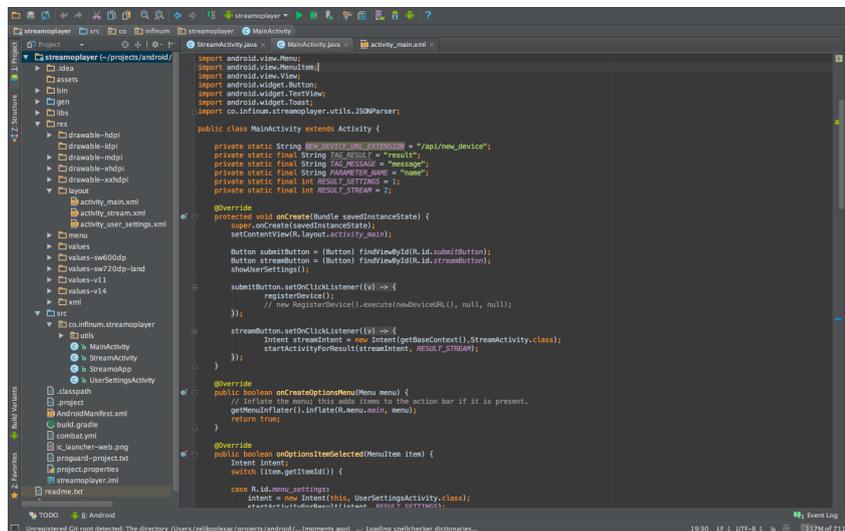


Figura 3: Android Studio IDE(PLESAC, 2013)

### 4.2.2 Linguagens de Programação

Na aplicação móvel foi utilizada a linguagem Java para desenvolvimento em Android (1.1), com XML para definição das interfaces gráficas. Esta foi escolhida devido a familiaridade do autor, somado ao incentivo da vasta utilização de aparelhos com base neste sistema operacional.

Na parte do servidor, a página *web* foi criada utilizando a API GoogleMaps, com pequenas porções de HTML. A comunicação cliente-servidor utiliza o protocolo JSON, devido sua simplicidade e leveza. A comunicação entre a *webpage* e o banco de dados é feita utilizando-se protocolo AJAX para chamadas assíncronas conforme as requisições de dados pela página. A linguagem PHP foi considerada para executar toda a lógica por trás do que é visível ao usuário, como a leitura e tratamento dos dados.

## 4.3 Metodologia de desenvolvimento

Nas primeiras fases de desenvolvimento do projeto, procurou-se compreender e utilizar o padrão internacional de medida IRI devido a sua aceitação na área. Todavia, foi visto que o conceito envolvido era de alta complexidade, necessitando de muito processamento e em alguns casos, de equipamentos adicionais. Dessa forma, optou-se pela análise dos dados puros da leitura, média e desvio padrão da mesma.

Foi notado através de testes práticos que a média não poderia ser utilizada, uma vez que as vibrações se situam em torno de zero e eventualmente geram valores idênticos em situações distintas. Essa teoria também eliminou a possibilidade de empregar o coeficiente de variação como métrica, dado que em momentos que a média atingia valores menores que 1, e representando o denominador da fórmula matemática, o resultado era amplificado para valores irrelevantes.

O desvio padrão por sua vez, demonstrou ser uma medida viável visto que representou satisfatoriamente e notavelmente os impactos recebidos no veículo. Para superfícies lisas com pouco ou nenhum defeito, valores baixos são lidos, e o contrário para regiões com deformidades, sendo sempre coerentes em magnitude com a força do impacto. Abaixo é demonstrado a leitura pura do acelerômetro junto ao desvio padrão em vermelho (Figura 4

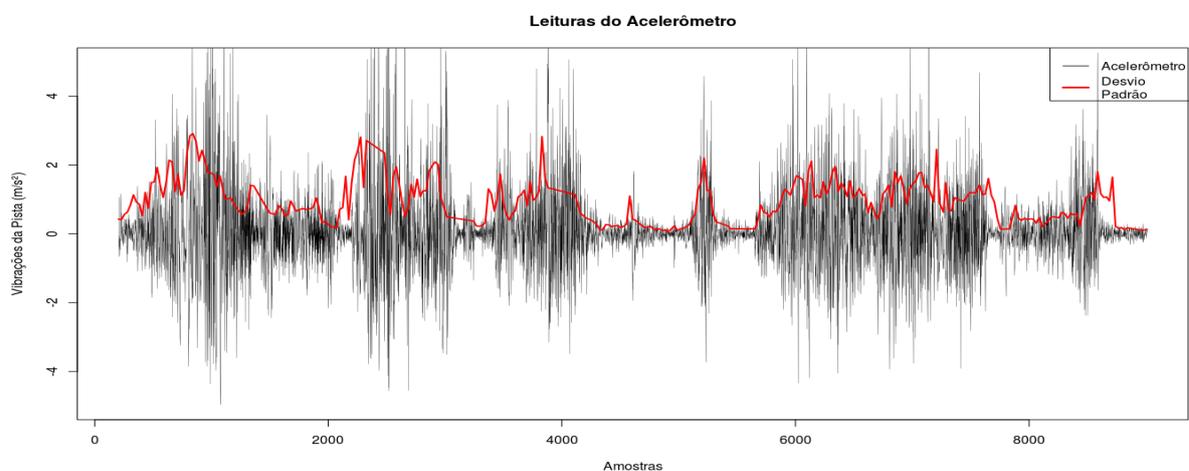


Figura 4: Leituras do acelerômetro e desvio padrão de uma pista real

Uma variável de grande importância para o algoritmo é a velocidade do automóvel, devido a sua influência direta na leitura (DU et al., 2014)(YI; CHUANG; NIAN, 2013). Em trabalhos prévios é demonstrado que o cálculo do IRI é feito apenas para velocidades superiores a 20km/h, e para velocidades inferiores alguns pesquisadores utilizam outras abordagens. O projeto apresentado aqui não irá trabalhar com esta métrica, mas ignorará leituras abaixo deste limiar, já que isto evita a leitura distorcida de arranques e alguns quebra-molas, que não compõem o escopo da pesquisa. Com o aumento da velocidade, os sensores captam valores maiores para o mesmo trecho ou deformidade. Esta não é uma

relação linear de natureza simples, logo, regras de proporções não podem ser utilizadas aqui.

Com o empecilho gerado pela interferência da velocidade nos resultados, procurou-se o uso de regressão linear para encontrar uma função que melhor representasse o trecho avaliado. Foi criado um *script* desenvolvido na ferramenta matemática R para o cálculo da regressão, utilizando a relação velocidade *versus* desvio padrão. Contudo, além do alto tempo de processamento para larga escala de dados, cada carro possui características próprias como: peso, amortecedores, vibração do motor, idade, entre outros. Isto impossibilita a utilização da regressão devido a natureza caótica das informações analisadas, não gerando tendência que pudesse descrever uma curva representativa dos dados.

Por fim, a ideia final foi a criação de um código capaz de analisar os dados em tempo real, utilizando o próprio carro e seu histórico recente como parâmetro de comparação, separados por faixas de velocidades de 10km/h cada. A saída do algoritmo é uma sinalização (*flag*) que obedece a seguinte classificação em suas respectivas cores como mostrado na tabela 1 a seguir:

Tabela 1: Flags e cores correspondentes.

| Flag | Qualidade        | Cor      |
|------|------------------|----------|
| 0    | Bom              | Verde    |
| 1    | Normal           | Amarelo  |
| 2    | Ruim             | Laranja  |
| 3    | Horrível         | Vermelho |
| 4    | Problema Pontual | Preto    |

Na figura 5 pode ser visto a tela da aplicação com as cores citadas em cada momento de classificação.



Figura 5: Fotos da tela do *smartphone* representando as qualidades em tempo real de diferentes pavimentos.

As próximas subseções são dedicadas a explicação mais detalhada do sistema desenvolvido, separados em blocos funcionais:

### 4.3.1 Coleta de Dados

Por ser uma aplicação *crowdsourcing* com diversos contribuidores, o modelo dos sensores utilizados dependem do aparelho que a aplicação está sendo executada. A coleta é realizada sempre que o veículo se encontra acima de 20km/h, eliminando assim a leitura de alguns quebra-molas e arrancadas. As informações de velocidade e localização são fornecidas pelo GPS do aparelho, que são consultados a cada coleta do acelerômetro (figura 6).

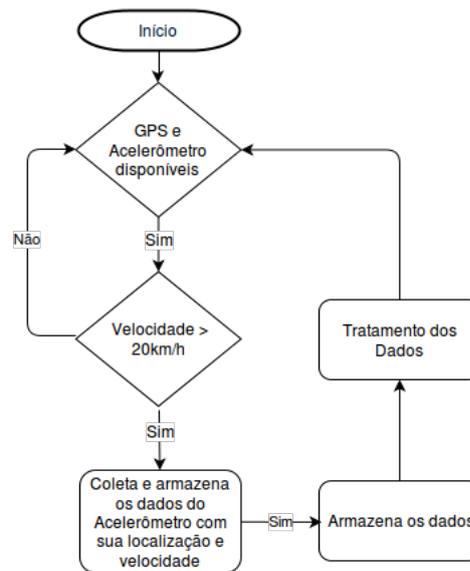


Figura 6: Fluxo de Coleta dos Dados

Como visto na seção 1.2, o acelerômetro capta a aceleração gravitacional da terra, sendo esta aproximadamente  $9,8 \text{ m/s}^2$  quando o aparelho está em repouso. Por questões práticas, é isolado a aceleração do aparelho descartando-se a gravidade. Para isto utiliza-se um filtro passa alta (DEVELOPER, 2016) fornecido pelo site oficial da Google de desenvolvimento *Android*, também explicado em (ENGINEERING, 2014).

Cada leitura gera um conjunto de informações, que contém as leituras dos três eixos (x, y e z) do acelerômetro, localização (latitude e longitude), o ângulo da direção do carro com o norte geográfico, a velocidade fornecida pelo GPS, a matriz de rotação utilizada na seção 4.3.2. Tais valores são então armazenados para processamento futuro.

### 4.3.2 Processamento

Neste bloco é onde toda a lógica de análise de qualidade se situa, realizando a leitura dos dados providos pela etapa de coleta, identificando os problemas, gerando as notificações (*flags*) e armazenando o novo conjunto de informações para serem enviadas ao servidor (seção 4.3.3).

Os dados armazenados da coleta são lidos sequencialmente para serem processados, e primeiramente realiza-se a rotação virtual dos eixos. Esta rotação utiliza o campo magnético da terra para encontrar o norte magnético, e com isso realizar os cálculos como se o celular estivesse com o eixo Z apontado perpendicularmente à terra. Em outras palavras, apontando para o teto do carro. Dessa forma, na prática, é feito a análise em cima da leitura do movimento vertical do carro.

Cada vez que se detecta um deslocamento maior que um metro, é realizado o cálculo do desvio padrão das leituras dos acelerômetros deste trecho, criando-se um “segmento de pista” ou “segmento representativo” daquele trecho (figura 7). Estes segmentos podem possuir comprimentos maiores que um metro dependendo da velocidade do carro e da atualização do GPS. As comparações para detecção da qualidade são realizadas entre as médias dos últimos segmentos com a leitura atual.

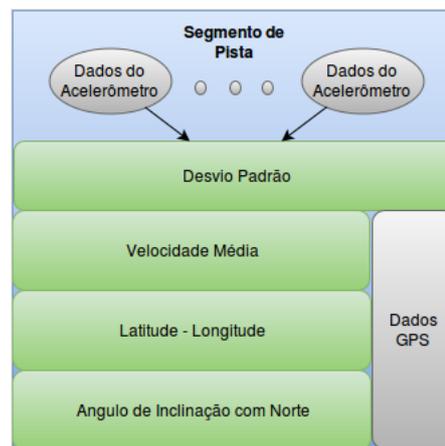


Figura 7: Elementos armazenados dentro de cada segmento de pista

Cada segmento representativo armazena em sua estrutura, além de outras informações, a velocidade média daquele trecho. Com esse dado é realizado a separação destes em intervalos de velocidade de 10km/h cada, inicializando a partir de 20km/h. Esta separação é necessária devido a influência da velocidade na magnitude da leitura dos dados. Desta forma, compara-se os dados com aqueles similares de forma a se considerar sempre os de “influência semelhantes”. As informações de um intervalo de velocidade só serão analisadas após obter um número mínimo de dados para o cálculo da média dos desvios padrões.

O algoritmo possui duas lógicas para classificação das qualidades e uma para detecção de buracos, que serão explicados nas subseções que se seguem. Vale citar que todas as lógicas possuem uma margem de erro devido a natureza desse problema. Cabe a etapa de conhecimento coletivo corrigir estas informações, trabalhando com as informações de muitos e não de um único usuário.

#### 4.3.2.1 Casos Extremos - Identificação de Trechos Bons e Péssimos

Acompanhado da separação de cada segmento em intervalos de velocidade, também armazena-se um conjunto de todas as leituras sem distinção de velocidade (figura 6) chamado “setor global”, que é utilizado para a identificação de trechos bons e horríveis.

Durante a fase experimental e análise dos dados foi notado que em trechos bons as leituras se localizavam em valores baixos (menores que 1, chamado de “LOWTHRESH”) e com vibrações baixas. O contrário foi notado para trechos péssimos, que possuem valores altos (maiores que 3, chamado de “HIGHTHRESH”) com grande distanciamento da média.

Com este resultado é possível saber de antemão se o veículo se encontra em um pavimento muito bom ou muito ruim. Quando a média geral das leituras atinge valores inferiores a “LOWTHRESH” ou superiores a “HIGHTHRESH”, o veículo se encontra em um trecho bom (verde) ou péssimo (vermelho) respectivamente.

Enquanto uma das duas condições forem satisfeitas, o vetor de velocidades distintas não será atualizado. Este vetor é dedicado apenas para a identificação dos casos intermediários, explicado a seguir.

#### 4.3.2.2 Casos Intermediários - Identificação de Trechos Médios e Ruins

Considerando que a aplicação já coletou informações iniciais suficientes de velocidade, o sistema procura identificar se os novos dados se diferem da média das últimas leituras para aquele intervalo de velocidade. Dados com grandeza maior que 50% da média são considerados picos, que podem representar um problema pontual, ou o início de uma sequência de problemas.

Após a identificação de um pico (condição de entrada), identifica-se se os trechos subsequentes também são picos, representando assim um pavimento ruim. A leitura de três não-picos subsequentes identificam a saída (condição de saída) de um trecho ruim ou a constatação de que possa ser apenas um pico isolado. Picos isolados podem representar buracos (seção 4.3.2.3) ou informações pontuais que serão descartadas. A figura 8 dedica-se a explicação desta ideia.

Sempre que uma condição de entrada é satisfeita, todos os dados serão armazenados separadamente até que uma condição de saída seja encontrada. Durante este processo, a média também não será atualizada. Se a média de desvios fosse atualizada, ela rapidamente subiria a ponto de não mais considerar picos como problemas, e assim o pavimento passaria a ser classificado como normal. A média das últimas leituras existe unicamente para identificar o limiar entre o que é médio e ruim, e criar uma margem de comparação.

Enquanto picos não são encontrados, o pavimento será considerado normal. Uma região com três ou mais picos próximos será considerada um pavimento ruim até que a condição de saída seja encontrada.

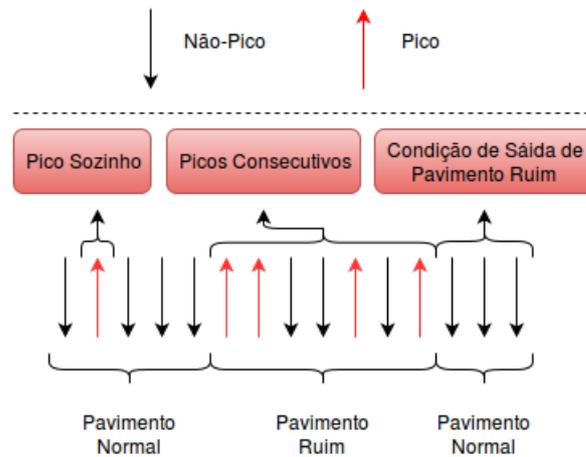


Figura 8: Exemplo de como RoadScan detecta qualidades intermediárias e suas condições de entrada e saída

Como visto na subseção anterior, valores inferiores a “LOWTHRESH” ou superiores a “HIGHTHRESH” representam pavimentos bons ou péssimos, respectivamente. Sendo assim, os casos intermediários situam-se entre essa faixa de valores, e são separados através da margem de comparação citada previamente, com a identificação sequencial de picos. Uma visão geral das últimas seções pode ser observada nas figuras 9 e 10.

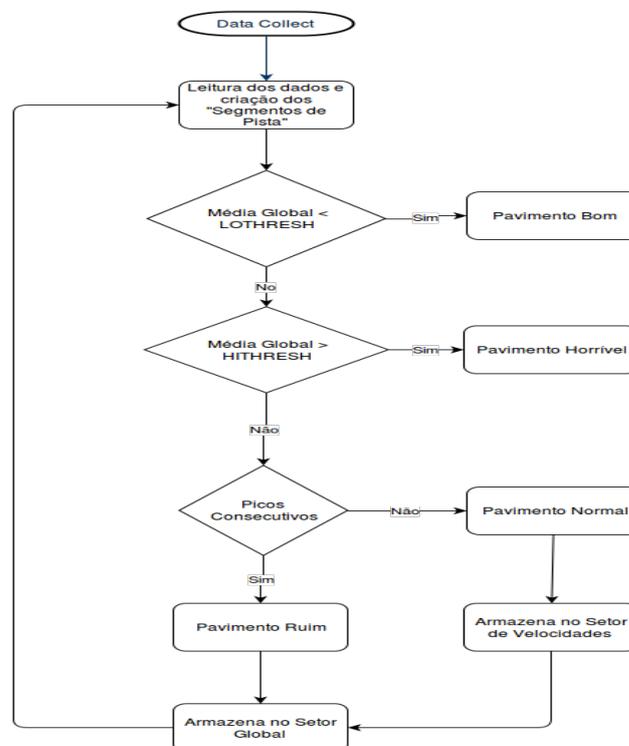


Figura 9: Fluxo de tratamento dos dados

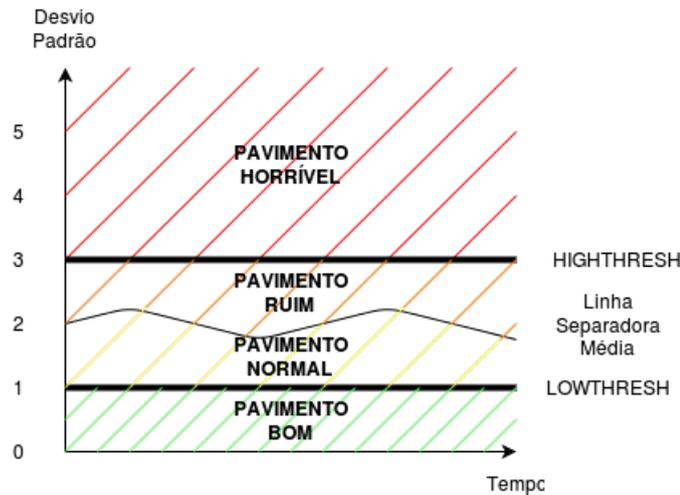


Figura 10: Categorização das quatro qualidades de pavimentos

#### 4.3.2.3 Caso Isolado - Identificação de Problemas Pontuais

Picos de magnitude maior que 100% em relação a média são considerados problemas pontuais. Um problema pontual é identificado posteriormente pelo sistema após a análise dos próximos trechos. Problemas pontuais podem representar buracos, bueiros (sumidouros), ou deformidades bruscas.

Independente da natureza real do problema este é identificado, armazenado, e possuirá um tratamento particular no servidor.

#### 4.3.3 Comunicação Cliente-Servidor

Cada segmento, após análise, receberá uma *flag* referente a sua qualidade (tabela 1). Os dados serão então depositados no banco de dados local do aparelho para envio ao servidor de armazenamento.

Uma linha de execução paralela se encarrega de checar o tamanho do banco periodicamente, e verifica se este possui tamanho mínimo suficiente para gerar um pacote de informações a serem enviadas. O envio das informações é feita utilizando-se o formato JSON devido a sua leveza e simplicidade.

A aplicação aguarda pela resposta de confirmação do servidor para que possa devidamente remover as informações enviadas do armazenamento local. Caso contrário, tentará novamente.

#### 4.3.4 Servidor

O servidor realiza a parte que será exibida ao usuário junto com o armazenamento dos dados. A este é encarregado as seguintes tarefas:

- Receber e armazenar os dados enviados pela aplicação *RoadScan* em um banco local;
- Processamento dos dados e aplicação do *crowdsourcing*;
- Desenhar as linhas com as devidas cores e informações na API *GoogleMaps*;
- Hospedar a *webpage* com o mapa de consulta.

O recebimento dos dados é realizado desmembrando o pacote JSON com as informações e as inserindo no banco de dados local da máquina. Caberá à etapa de processamento ler estes dados conforme as requisições dos usuários, e realizar limpeza no banco reduzindo a quantidade de informações. Algoritmos foram desenvolvidos para o desenhos das retas e para a limpeza do banco, visando a melhora do tempo de carregamento da página. As etapas de maior relevância serão explicadas nas seções seguintes.

#### 4.3.4.1 Leitura dos Dados e Desenho das Linhas

A cada acesso a página é feito uma requisição ao servidor para o desenho da tela, filtrando primeiramente apenas os pontos contidos na tela visualizada. O conjunto de pontos retornados pelo banco carrega consigo suas localizações, *flags* de qualidade, e angulação do movimento do automóvel em relação ao norte geográfico.

O processamento é realizado seguindo a ordem crescente de *id's* das tuplas recebidas do banco. Uma iteração representa um segmento representativo mencionado em 4.3.2, que são as linhas do mapa. A cada linha é necessário procurar por dados em localizações próximas, que representam informações de outras contribuições para o mesmo trecho de pavimento.

A procura é feita em forma de retângulos que contêm a linha analisada, como visto na figura 11. É realizada a consulta das *flags* destes pontos e calculada a média destes valores para representar o valor final que representará a cor da linha.



Figura 11: Retângulos de busca para agrupar dados próximos.

A quantidade de *id's* não sequenciais presentes em um retângulo representa con-

tribuições distintas para aquele trecho. Estas contribuições espelham a fidelidade da informação daquela linha. Apenas linhas com uma fidelidade mínima serão exibidas no mapa, evitando assim informações pobres e erros de precisão do GPS que usualmente produzem pontos incoerentes com as vias.

Um obstáculo foi encontrado em casos de vias de mão dupla (dois sentidos de tráfego), uma vez que o retângulo poderia abranger informações do outro lado da pista. Por esse motivo é enviado e armazenado a informação de angulação do movimento do carro em relação ao norte. Com este dado é possível diferenciar dados que apontam em direções aproximadamente opostas e realizar a discriminação destes.

A cada linha é atribuído uma pequena janela de informações acessada via clique contendo sua qualidade, número de avaliações e a fidelidade em uma representação de estrelas, como visto na figura 12.

#### 4.3.4.2 Tratamento de Problemas Pontuais

Como mencionado na seção 4.3.2.3, problemas pontuais possuem um tratamento semelhante ao realizado em linhas. No lugar de retângulos é utilizado um círculo para a procura dos problemas. A região circular procura por outras incidências do problema e caso obtenha um numero suficiente de ocorrências, é então desenhado um pequeno círculo de cor preta para identificá-lo.

#### 4.3.4.3 Condensamento das Informações

Com o decorrer do tempo o banco tende à aumentar o volume de informações, isso implica em mais tempo para execução do algoritmo de processamento, e mais recursos necessários para o armazenamento destes dados.

Para promover um acesso mais agradável à página e poupar o servidor de processamento desnecessário, foi criada uma lógica de condensamento dos dados. Acompanhado dessa ideia também é estabelecido um nível de zoom mínimo para exibição dos dados.

Cada linha possui um ponto que indica o início e o fim da mesma. Todos os pontos encontrados no retângulo referente a reta geram um valor final de qualidade. Em tempo de execução, o algoritmo armazena o resultado no ponto de início, junto a quantidade de avaliadores para identificação da fidelidade. Quando esta situação ocorre também é realizada a limpeza do banco das tuplas sintetizadas, restando apenas um ponto representativo.

Com a aplicação deste algoritmo uma região totalmente mapeada terá uma quantidade máxima de pontos para representá-la, e conseqüentemente um tempo aproximadamente determinado até sua exibição na tela.



Parte III

Resultados



RoadScan após três meses em execução com múltiplos usuários apresentou resultados promissores. Observando as linhas geradas e desenhadas no mapa, nota-se uma grande semelhança com o estado real das pistas (figura 12). Isto demonstra que a aplicação dos retângulos de busca e *crowdsourcing* obtiveram êxito em produzir o resultado final. Além disso, a resposta em tempo real na tela do aplicativo também refletiu de maneira satisfatória as condições da pista trafegada.



Figura 12: Mapa resultante referenciando as qualidades e os trechos reais.

Também foi constatado que, toda vez que o RoadScan classifica um pavimento erroneamente, este erra por apenas um nível de diferença. Isto significa que o sistema dificilmente irá produzir respostas discrepantes em relação as ideais, reforçando sua acurácia. Em consequência disso, pode-se também aspirar por resultados coerentes e precisos na página online.

A linha separadora média demonstrou ser um método eficiente para tratar as pequenas diferenças comportamentais de cada veículo. Isto se deve ao fato de que cada carro é comparado consigo mesmo e seu próprio histórico de leituras. Diferentes modelos de carro e posicionamento dos *smartphones* geram diferentes leituras no acelerômetro. Contudo, como o sistema utiliza de uma relação entre valores para identificar os picos, a magnitude dos valores se mantém. Se por ventura o aparelho for reposicionado, a média

de comparação logo irá se adaptar para a nova posição, ajustando o sistema.

A utilização do padrão JSON para envio de informações apresentou-se efetivo devido sua simplicidade de entendimento, manuseio e baixo *overhead*.

Leituras indesejáveis na parte da aplicação são corrigidas na parte servidora. Ambos trabalham em conjunto, e espera-se que o *crowdsourcing* suprima tais erros conforme novas informações são coletadas. É esperado que leituras errôneas representem uma pequena porção que não influenciará o resultado integralmente. Além disso, conta-se também com o nível de fidelidade para filtrar informações aleatórias e inadequadas.

O algoritmo de busca desenvolvido, utilizando retângulos contendo as linhas, atendeu as expectativas. Como estes são em grande parte paralelos as ruas, os mesmos conseguem captar bem as informações próximas desejáveis. A utilização do ângulo de inclinação do movimento conseguiu eliminar por completo o empecilho gerado por vias de mão dupla. Esta abordagem foi capaz de evitar que dados próximos fossem incorretamente agrupados, situação esta que ocorre na maioria das pistas.

O condensamento das informações em um único ponto reduziu drasticamente o tempo de carregamento da página, mostrando-se além de eficiente, indispensável. Com o aumento do número de informações coletadas, não só o tempo de carregamento, mas também o tamanho do banco cresceriam indefinidamente. Tal comportamento iria esgotar os recursos além de tornar o uso da página desagradável e impraticável.

# Conclusões

Este trabalho apresentou RoadScan, uma abordagem distintiva com base em um aplicativo Android para análise de qualidade de pavimentos. Um sistema composto por uma aplicação com capacidade de resposta em tempo real e uma página *web* capaz de apresentar os resultados mais atuais usufruindo do conhecimento coletivo para prover dados mais precisos. A utilização de *smartphones* na solução de paradigmas cotidianos é um passo em direção a inovação e praticidade, além de apresentarem recursos competentes e preço acessível. Seu uso está aumentando a cada dia e atinge escala mundial, provendo assim o potencial necessário para manter as informações sempre atualizadas. Soluções utilizando dispositivos móveis é uma tarefa desafiadora, uma vez que diversas variáveis devem ser levadas em consideração. Cada automóvel possui suas próprias características, e assim como pavimentos, suas propriedades e condições se alteram com o tempo. Leituras realizadas pelos sensores são apenas distúrbios, possuindo assim uma grande variedade de possibilidades para sua fonte. Batimento de portas, freadas, quebra-molas, interação do usuário entre outros, produzem medidas significantes de sinal, e uma maneira de filtrar o que é desejável deve ser implementado. O posicionamento do aparelho interfere drasticamente nas leituras, por isso, técnicas como rotação virtual dos eixos e utilização de magnitude para identificação de picos foram utilizadas neste trabalho. Com os bons resultados aqui relatados, acredita-se que RoadScan possui o potencial para ser uma ferramenta confiável para o monitoramento de qualidade de pavimentos. Apesar de apresentar carência em precisão, o mesmo provê um resultado aceitável cumprindo seu objetivo: uma solução acessível para supervisão de pavimentos acompanhado de consciência social.



## 5 Trabalhos Futuros

RoadScan apresentou um decente resultado para sua primeira versão, de qualquer forma, este certamente precisa de melhorias para prosseguir. Embora algumas constantes tenham sido definidas baseadas na média do comportamento detectado nos experimentos, estes valores precisam ser escolhidos com mais cautela. Acredita-se que HIGHTRESH e LOWTHRESH são valores essenciais para que o sistema abranja outros tipos de veículos, como motos, carros de grande porte, ônibus, entre outros. Repetir os mesmo testes para cada um destes tipos poderia criar um conjunto de constantes que cobriria uma boa parte dos modelos, mas isso não seria o ideal. Levando em consideração que carros envelhecem e podem possuir partes mecânicas em mal estado, o ideal seria utilizar uma máquina de aprendizado em cada execução para que o sistema se adapte automaticamente em qualquer tipo de veículo ou posicionamento do aparelho. Com o passar do tempo, perfis de automóveis seriam criados, calibrados e identificados através de metodologias como os modelos SVM (MEYER, 2014). A precisão dos sensores GPS também apresentou alguns problemas na parte do servidor. Uma vez que o desenho das linhas dependem estritamente da localização dos segmentos de pista, se o GPS prover informações incorretas, problemas nos desenhos e agrupamento das informações são gerados. O nível de fidelidade é capaz de filtrar algumas dessas informações mal localizadas, mas não todas. Alguns dados também podem ser inadequadamente agrupados com outros segmentos gerando problemas na consistência dos resultados. A solução seria elaborar uma maneira de calibrar o GPS previamente, e aceitar apenas informações com um certo nível de confiança e precisão. A implementação de uma marca temporal em cada informação também é relevante. Será futuramente necessário verificar e limpar informações antigas e obsoletas. Além destes tópicos principais, algumas melhorias no desempenho, consumo energético da bateria do aparelho e aparência da aplicação são convenientes.



# Referências

- ADLINGE, S. S.; GUPTA, P. a. K. Pavement Deterioration and its Causes. Mechanical & Civil Engineering, p. 9–15, 2009. ISSN 2278-1684. Disponível em: <[www.iosrjournals.org](http://www.iosrjournals.org)>. Citado 2 vezes nas páginas 23 e 35.
- ANDROID, G. Google Android. 2016. Disponível em: <[https://www.android.com/intl/pt-BR\\_br/](https://www.android.com/intl/pt-BR_br/)>. Acesso em: 18 jan 2016. Citado na página 27.
- ANDROID Open Source Project. 2016. Disponível em: <<https://source.android.com/source/>>. Acesso em: 14 jun 2016. Citado na página 27.
- ARHIN, S. a. et al. Predicting pavement condition index using international roughness index in a dense urban area. v. 5, 2015. Citado na página 24.
- BARTHOLOMEU, D. B.; Caixeta Filho, J. V. Impactos econômicos e ambientais decorrentes do estado de conservação das rodovias brasileiras: um estudo de caso. Revista de Economia e Sociologia Rural, v. 46, n. 3, p. 703–738, 2008. ISSN 0103-2003. Citado na página 35.
- CHEN, K. et al. Road condition monitoring using on-board three-axis accelerometer and GPS sensor. Proceedings of the 2011 6th International ICST Conference on Communications and Networking in China, CHINACOM 2011, n. 2009, p. 1032–1037, 2011. Citado na página 24.
- CHUGH, G.; BANSAL, D.; SOFAT, S. Road Condition Detection Using Smartphone Sensors: A Survey. v. 7, n. 6, p. 595–602, 2014. Citado 2 vezes nas páginas 23 e 31.
- DEVELOPER, A. SensorEvent. 2016. Disponível em: <<https://developer.android.com/reference/android/hardware/SensorEvent.html>>. Acesso em: 30 maio 2016. Citado na página 43.
- DEVELOPERS, G. Sensors Overview. 2016. Disponível em: <[https://developer.android.com/guide/topics/sensors/sensors\\_overview.html](https://developer.android.com/guide/topics/sensors/sensors_overview.html)>. Acesso em: 06 jun 2016. Citado na página 28.
- DOUANGPHACHANH, V.; ONEYAMA, H. A Study on the Use of Smartphones for Road Roughness Condition Estimation. ... for Transportation Studies, v. 9, n. 2007, 2013. Disponível em: <<http://east.info/on-line/proceedings/vol9/PDF/P297.pdf>>. Citado na página 24.
- DOUANGPHACHANH, V.; ONEYAMA, H. Exploring the Use of Smartphone Accelerometer and Gyroscope to Study on the Estimation of Road Surface Roughness Condition. Informatics in Control, Automation and Robotics (ICINCO), 2014 11th International Conference on, p. 783–787, 2014. Citado na página 24.
- DU, Y. et al. Measurement of International Roughness Index by Using Z-Axis Accelerometers and GPS. Mathematical Problems in Engineering, v. 2014, p. 1–10, 2014. ISSN 1024-123X. Disponível em: <<http://www.hindawi.com/journals/mpe/2014/928980/>>. Citado na página 41.

ENGINEERING, A. Pothole Detection System using Machine Learning on Android. v. 4, n. 7, p. 360–364, 2014. Citado na página 43.

ERIKSSON, J.; GIROD, L.; HULL, B. The pothole patrol: using a mobile sensor network for road surface monitoring. ... conference on Mobile ..., p. 29, 2008. Disponível em: <<http://dl.acm.org/citation.cfm?id=1378600.1378605%delimitador%026E30F%nhhttp://dl.acm.org/citation.cfm?id=1378605>>. Citado na página 31.

FALLIS a.G. Classificação de Qualidade de Pistas com Vistas ao Desempenho de Pneus. *Journal of Chemical Information and Modeling*, v. 53, n. 9, p. 1689–1699, 2013. ISSN 1098-6596. Citado na página 35.

GERENCIAL, R. Relatório gerencial. 2015. Citado na página 23.

GOOGLE Opinion Rewards. 2016. Disponível em: <[https://play.google.com/sRtore/apps/details?id=com.google.android.apps.paidtasks&hl=pt\\_BR](https://play.google.com/sRtore/apps/details?id=com.google.android.apps.paidtasks&hl=pt_BR)>. Acesso em: 06 jun 2016. Citado na página 29.

HALL, D.; LEWIS, J.; WOLFE, S. Pavement Condition Surveys – Overview of Current Practices. v. 19716, n. 302, 2013. Citado na página 23.

HILDENBRAND, J. What is Android? 2015. Disponível em: <<http://www.androidcentral.com/what-android>>. Acesso em: 06 jun 2016. Citado na página 27.

IDC. Smartphone OS Market Share, 2015 Q2. 2015. Disponível em: <<http://www.idc.com/prodserv/smartphone-os-market-share.jsp>>. Acesso em: 20 jan 2016. Citado 2 vezes nas páginas 27 e 35.

IDEA. What is Crowdsourcing? And how does it apply to outreach? 2016. Disponível em: <<http://www.idea.org/blog/2013/02/19/what-is-crowdsourcing-and-how-does-it-apply-to-outreach/>>. Acesso em: 06 jun 2016. Citado na página 28.

KICKSTARTER. Kickstarter. 2016. Disponível em: <<https://www.kickstarter.com>>. Acesso em: 06 jun 2016. Citado na página 29.

KLAUBERT, E. C. Highway Effects on Vehicle Performance. n. January, p. 77, 2001. ISSN 02773775. Citado na página 23.

MEDNIS, A. et al. Real time pothole detection using Android smartphones with accelerometers. *2011 International Conference on Distributed Computing in Sensor Systems and Workshops, DCOSS'11*, 2011. Citado na página 32.

MELOROSE, J.; PERROY, R.; CAREAS, S. A Synopsis on the Current Equipment Used for Measuring Pavement Smoothness. *Statewide Agricultural Land Use Baseline 2015*, v. 1, 2015. ISSN 1098-6596. Citado na página 23.

MELOROSE, J.; PERROY, R.; CAREAS, S. Automated Collection Of Road Condition Data With Smartphones. *Statewide Agricultural Land Use Baseline 2015*, v. 1, 2015. ISSN 1098-6596. Citado na página 31.

MEYER, D. Support vector machines: the interface to libsvm in package e1071. ... *Systems and their ...*, v. 1, p. 1–8, 2014. ISSN 1613-9011. Disponível em: <[http://ieeexplore.ieee.org/xpls/abs{\\\\_}all.jsp?arnumber=708](http://ieeexplore.ieee.org/xpls/abs{\\_}all.jsp?arnumber=708)>. Citado na página 57.

- MOHAN, P.; PADMANABHAN, V. N.; RAMJEE, R. TrafficSense : Rich Monitoring of Road and Traffic Conditions using Mobile Smartphones. The 6th ACM Conference on Embedded Networked Sensor Systems, p. 1–29, 2008. Disponível em: <ftp://ftp.research.microsoft.com/pub/tr/TR-2008-59.pdf>. Citado na página 31.
- MOHAN, P.; RAMJEE, R.; PADMANABHAN, V. N. Nericell: rich monitoring of road and traffic conditions using mobile smartphones. 6th ACM conference on Embedded network sensor systems (SenSys '08), p. 323–336, 2008. Disponível em: <http://research.microsoft.com/pubs/78568/Nericell-Sensys2008.pdf>. Citado na página 31.
- OFFICE, B. M. Street Bump. 2016. Disponível em: <http://www.streetbump.org/about>. Acesso em: 06 jan 2016. Citado na página 31.
- PFLEEGER, S. Everything You Wanted to Know about the IRI, But Were Afraid to Ask. IEEE Security, v. 4, n. 3, p. 5–5, 2006. Citado na página 24.
- PLESAC Željko. Android Studio vs. Eclipse. 2013. Disponível em: <https://infinum.co/the-capsized-eight/articles/android-studio-vs-eclipse-1-0>. Acesso em: 14 jun 2016. Citado na página 40.
- ROADROID. Roadroid. 2016. Disponível em: <http://www.roadroid.com/>. Acesso em: 07 jun 2016. Citado 2 vezes nas páginas 31 e 32.
- SAYERS, M. W.; GILLESPIE, T. D.; PATERSON, W. D. Guidelines for conducting and calibrating road roughness measurements. [s.n.], 1986. 98 p. ISBN World Bank Technical Paper Number 46. Disponível em: <http://trid.trb.org/view.aspx?id=278294>. Citado na página 24.
- SUNG, D. What is wearable tech? Everything you need to know explained. 2015. Disponível em: <http://www.wearable.com/wearable-tech/what-is-wearable-tech-753>. Acesso em: 06 jun 2016. Citado na página 27.
- WHAFF. WHAFF Rewards. 2016. Disponível em: [https://play.google.com/store/apps/details?id=com.whaff.whaffapp&hl=pt\\_BR](https://play.google.com/store/apps/details?id=com.whaff.whaffapp&hl=pt_BR). Acesso em: 14 jun 2016. Citado na página 29.
- WIKIPEDIA. Wikipedia. 2016. Disponível em: <https://pt.wikipedia.org/>. Acesso em: 06 jun 2016. Citado na página 29.
- YI, C.-W.; CHUANG, Y.-T.; NIAN, C.-S. Toward crowdsourcing-based road pavement monitoring by mobile sensing technologies. Intelligent Transportation Systems, v. 16, n. 4, p. 1905–1917, 2013. Disponível em: [http://ieeexplore.ieee.org/xpls/abs{\\\\_}all.jsp?arnumber=7056](http://ieeexplore.ieee.org/xpls/abs{\\_}all.jsp?arnumber=7056). Citado na página 41.