

CURSO DE CIÊNCIA DA COMPUTAÇÃO

Lauro Tremea Culau

**SISTEMA DE MONITORAMENTO DE ESTACIONAMENTOS UTILIZANDO
DADOS DE GPS**

Santa Cruz do Sul

2016

Lauro Tremea Culau

**SISTEMA DE MONITORAMENTO DE ESTACIONAMENTOS UTILIZANDO
DADOS DE GPS**

Trabalho de Conclusão apresentado ao Curso de Ciência da Computação da Universidade de Santa Cruz do Sul, como requisito parcial para a obtenção do título de Bacharel em Ciência da Computação.

Orientador: Prof. Dr. Leonel Pablo Tedesco

Santa Cruz do Sul
2016

RESUMO

Com o crescimento populacional das últimas décadas, aliado à migração de pessoas do interior para as grandes e médias cidades, os problemas de mau planejamento das cidades começam a aparecer. Os principais, e que são os vistos por primeiro, se relacionam à mobilidade urbana. Como principais exemplos, pode-se citar a perda de qualidade e problemas para gerenciamento de transportes públicos, como ônibus e metrô. Aliados a eles, o trânsito, em si, se torna outro inimigo da população, com um considerável aumento no tráfego de veículos e nos congestionamentos. Com estes problemas acontecendo, tudo o que está ligado a eles também começa a se tornar problema para a população, e entre eles se destaca a dificuldade de se encontrar vagas de estacionamento livres e de se estacionar um veículo, seja em estacionamentos públicos ou privados, principalmente em locais com maior concentração de pessoas, como centro de cidades, áreas universitárias ou industriais. Mesmo em lugares privados, que possuem controle de quem pode acessar o local, é possível de se encontrar problemas de falta de vagas para estacionar. Utilizando esta premissa de que encontrar uma vaga para estacionar está se tornando cada vez mais complicado, aliada ao aumento do uso de dispositivos móveis, e a partir da utilização dos dados do *Global Positioning System (GPS)*, o presente trabalho apresenta uma ferramenta para monitorar estacionamentos e que detecta o momento em que alguém estaciona o carro e torna esta informação disponível para qualquer motorista que possua o aplicativo, exibindo a ele um mapa com as áreas onde há a maior concentração de veículos, ajudando o usuário na definição do local com a maior probabilidade de se encontrar uma vaga para estacionar. Como conclusão, foi possível de analisar que o GPS possui grande variação e imprecisão nos dados, que pode fornecer localização incorreta conforme o local, fazendo o aplicativo funcionar incorretamente. Como os resultados obtidos foram apenas através de testes realizados, não obteve-se dados quantitativos e qualitativos para ter-se uma análise de uso real do aplicativo. Porém, os dados de testes mostraram que a ferramenta funciona de maneira correta e pode ser aplicada para monitorar estacionamentos. Além disso, como há grande pesquisa nesta área de monitoramento, é possível melhorar esta ferramenta adicionando novas formas de monitoramentos, tornando o aplicativo mais confiável.

Palavras-chave: Estacionamentos, GPS, monitoramento, mobilidade urbana.

ABSTRACT

With the population's growth over the last decades, combined with people moving from countryside to big cities, the problems related to the poor urban planning start to show up. The main ones, or at least, the ones which are seen for first, are related to urban mobility. As example, it is possible to cite the lack of quality and management in the urban's transportation systems. With these issues, the traffic itself becomes also a big problem, with a considerable increasing in the number of vehicles, which raise the traffic jams. Everything that is related to them starts to collapse as well, becoming real problems to people. A great example is the parking system issue. It is becoming harder and harder to find a free space to park, especially in places where there is a huge concentration of people, as universities, downtown, or around industrial areas. Even private parking lots, where they have the control of people that have access to them, suffer from this lack of free parking spaces. Using this conception, that it is becoming harder to find a place to park, combined to the increasing in the use of smartphones and by using the Global Positioning System (GPS), this article presents a tool to monitoring parking lots which detects the moment when someone parks the car and makes this information available through a map to any driver who has the application. The map shows the areas with the biggest concentration of cars, helping the user in defining the area with the highest probability to find a place to park the car. As conclusion, during the development and testes it was able to check that GPS provides wrong data in some cases, depending on the location, which can make the app to read the wrong location and not working properly. As the collected data was only test-based, there is no real results or analisis of data. However, the data colected during the tests showed that the app works as expected and can be applied to control parking lots. In addition, the tool can be improved by adding another tipos of monitoring and analisis of data to make it more reliable.

Keywords: Parking, GPS, monitoring, urban mobility.

LISTA DE ILUSTRAÇÕES

Figura 1 Constelação de satélites de GPS na órbita do Planeta Terra.....	13
Figura 2 Funcionamento do sistema de GPS	14
Figura 3 Arquitetura do trabalho de Zheng et al. (2010)	22
Figura 4 Fluxo de dados entre dispositivo e servidor.....	24
Figura 5 Gráfico com a distribuição de uso de cada versão do Android.....	26
Figura 6 Fluxograma da obtenção do estado atual do dispositivo	30
Figura 7 Área em que é feita a detecção de estacionamento.....	32
Figura 8 Exemplo de funcionamento do agrupamento de marcadores com níveis de zoom diferentes	35
Figura 9 Tabelas do banco de dados para armazenar informações de preenchimento dos estacionamentos.....	36

LISTA DE TABELAS

Tabela 1 Comparativo de trabalhos relacionados.....	23
---	----

SUMÁRIO

1	INTRODUÇÃO	7
2	REFERENCIAL TEÓRICO	10
2.1	Estacionamentos	10
2.2	<i>Global Positioning System (GPS)</i>	11
2.3	Velocidade	15
2.3.1	Velocidade Média	15
2.3.2	Velocidade Instantânea	15
2.4	Aceleração	17
3	TRABALHOS RELACIONADOS	19
4	SOLUÇÃO DESENVOLVIDA	24
4.1	Desenvolvimento	25
4.1.1	Estacione UNISC	25
4.1.2	Servidor	35
4.2	Validação e Testes	37
5	CONSIDERAÇÕES FINAIS	39
	REFERÊNCIAS	42

1 INTRODUÇÃO

A mobilidade é uma condição importante para a qualidade de vida de um indivíduo. A interação social também pode ser facilitada pela habilidade de uma pessoa se mover (RIBEIRO et al., 2007). Uma forma prática de locomoção ocorre através da utilização de automóveis, pois alguns de seus benefícios estão ligados à comodidade de não depender de transporte público e poder gerenciar horários de chegadas e saídas, oferecendo ao usuário uma maior liberdade e independência (MOSCARELLI, 2003).

Em grandes cidades, a rapidez de locomoção de automóveis não é muito maior que a dos transportes públicos, por causa dos constantes congestionamentos que ocorrem em suas vias. Ainda assim, este tipo de meio de transporte pode ser considerado rápido. Um importante problema associado à popularidade de automóveis são os congestionamentos e falta de vagas em estacionamentos (PAIVA, 2008).

Segundo Luz (1997), como as cidades estão crescendo em termos populacionais, são cada vez mais notáveis os problemas relacionados aos estacionamentos. Como o autor cita, mesmo que o total de vagas de estacionamentos aumente, esse aumento não é no mesmo ritmo no qual as cidades se expandem. Outro fator que influencia este crescimento inferior, é o fato de que os investimentos em estacionamentos são normalmente mais reduzidos do que os em infraestrutura viária.

Dessa forma se torna mais comum para o usuário encontrar estacionamentos totalmente preenchidos ao tentar estacionar o seu carro. Tal fato é comum tanto em locais públicos quanto em estacionamentos privados. Nesta busca por local para estacionar é gasto um tempo considerável para se encontrar uma vaga e, estando longe do local de destino, se perde muito tempo também de deslocamento até o local desejado. Adicionalmente, uma quantidade de combustível é gasta até se conseguir estacionar o automóvel, sendo possível o usuário ficar preso em algum pequeno congestionamento com outras pessoas que também estão à procura de uma vaga para o seu veículo (ARNOTT; INCI, 2005).

Encontrar vagas de estacionamento tem se tornado algo cada vez mais complicado na maioria das cidades, seguindo o exemplo citado por Luz (1997), onde o autor apresenta os problemas relacionados a estacionamentos na Universidade Federal de Santa Catarina (UFSC). Tomando a Universidade de Santa Cruz do Sul (UNISC) como exemplo, a falta de vagas pode ser notada pelo grande aumento de carros estacionados nos canteiros de grama, em uma das faixas da via principal, e estacionados de forma equivocada.

A UNISC vem ampliando as vagas dentro do campus. Porém, com o aumento de alunos, funcionários e docentes a cada ano, as vagas mais próximas aos blocos acabam se esgotando rapidamente, especialmente no período da noite. Além disso, há estacionamentos que são mais distantes ou mal iluminados, o que faz com que as pessoas pouco os utilizem. Com esse exemplo, que Luz (1997) escreve e que já foi citado, é possível de observar na prática que o crescimento populacional é maior do que a infraestrutura existente.

A justificativa acadêmica para a pesquisa ser na área de monitoramento de estacionamentos está baseada no fato de existir um grande número de publicações relacionadas aos temas de controle de tráfego e de estacionamentos, principalmente utilizando processamento de imagens para a detecção, como por exemplo:

- Liu, Mohandes e Deriche (2013) citam o grande aumento no número de veículos em meio às cidades, e que é necessário o desenvolvimento de um sistema inteligente de estacionamento para ajudar os motoristas a encontrarem lugares próximos para estacionar. Para isso, ele utiliza um sistema baseado em imagens para detectar espaços vagos dentro de um estacionamento.
- Zhang e Yu (2014) que aplicam um sistema de estacionamento para uma universidade que busca melhorar o espaços de estacionamento e reduzir o tempo perdido na busca pelas vagas. Além deles,
- Neto et al. (2015) e Jayasydha, Karunamoorthy e Sureshkumar (2015) também apresentam trabalhos com análise de imagens, e mostram que há estudos na área de monitoramento de estacionamentos e que embasam a ideia do presente trabalho, principalmente com o uso de imagens.

Porém, como o processamento de imagem possui um custo computacional bem mais elevado, se comparado à utilização de *Global Positioning System* (GPS), e também por não se possuir câmeras adequadas e uma estrutura computacional em tempo real para processar as imagens e obter-se uma visão de estado de preenchimento dos estacionamentos, foi decidido a utilização de dados de GPS, pois assim o próprio dispositivo móvel do motorista calcula seu status e salva os dados em um servidor, que ficará disponível em tempo real para os usuários verificarem onde há a maior probabilidade de encontrarem uma vaga de estacionamento.

Desta forma, foi preciso pesquisar por diferentes trabalhos que utilizem GPS como método principal de obtenção de dados e estes utilizados para monitoramento de estacionamentos e movimentos, onde a maioria deles utiliza o GPS para obter dados de deslocamento e posição

dos veículos e a partir disso, analisá-los. Os trabalhos serão apresentados detalhadamente no capítulo de trabalhos relacionados.

O grande problema apresentado por estes trabalhos e que, unidos com a proposta definida, foi que para obter o estado de estacionado é necessário primeiramente definir se o objeto, no caso do trabalho, um veículo, está realmente estacionado, e não apenas parado para alguém entrar no carro, ou então parado na sinaleira. A partir disso definiu-se que o objetivo do trabalho foi de desenvolver uma aplicação para definir se um veículo está realmente estacionado e o local onde se encontra, baseando-se em informações do dispositivo do motorista, neste caso, o GPS, e dados de outros sensores provenientes do aparelho móvel.

Com isso, o trabalho desenvolvido se diferencia, pois é obtido o estado específico do usuário, de estacionado ou não. Além de apresentar os dados em um mapa com o preenchimento de cada área de estacionamento ajudando aos usuários em definir uma área para estacionar. Além de obter informações relevantes e úteis sobre o preenchimento de estacionamentos para histórico.

Desta maneira, juntamente com o objetivo descrito acima, a proposta do projeto foi de estudar como a computação pode ajudar na melhoria e na organização do trânsito, com foco nas áreas de estacionamentos, mostrando qual a situação de cada uma delas. Esta informação pode auxiliar na tomada de decisão do motorista sobre o local com a maior probabilidade encontrar uma vaga para estacionar o automóvel, evitando gastos extras de combustível ao rodar procurando vagas livres. Além disso, é possível fazer com que a própria organização proprietária das vagas os use para ter uma melhor análise de possíveis locais para colocar novas vagas de estacionamento.

O restante do trabalho está dividido da seguinte maneira: No Capítulo 2 estão descritos termos e parâmetros padrão utilizados no trabalho e que facilitarão o entendimento do mesmo. No Capítulo 3 é feita uma abordagem de trabalhos relacionados ao tema proposto e feita uma comparação entre ambos. O Capítulo 4 apresenta os passos do desenvolvimento do trabalho, além dos testes e validações. Os resultados, problemas encontrados, assim como possíveis trabalhos futuros e considerações finais são apresentadas no Capítulo 5, e por fim as referências utilizadas no trabalho.

2 REFERENCIAL TEÓRICO

Nesta seção serão descritos alguns termos, sistemas e tecnologias utilizados para a escrita do trabalho e que ajudará no entendimento do mesmo. Primeiramente, como primeira seção, é apresentada uma definição geral de estacionamentos, pois estes serão usados como o ambiente a ser analisado.

Seguindo o capítulo, a segunda seção apresenta características e o funcionamento do GPS, que será a ferramenta utilizada para a captura dos dados de localização do veículo. E, por último, nas duas seções finais, são descritos os termos de velocidade e aceleração, que estão relacionados ao deslocamento, que serão utilizados para obter-se o status do veículo e analisar o histórico de movimento.

2.1 Estacionamentos

Segundo o Código de Trânsito Brasileiro (DENATRAN, 2008), a definição de estacionamento é o ato um veículo ficar parado por um tempo superior ao necessário para o embarque ou o desembarque de passageiros em um mesmo local.

Os estacionamentos são parte fundamental das cidades, desde centros até áreas mais afastadas, visto que e a sua demanda, apesar de ser maior em centros de cidades, pode acontecer também em locais fora dos centros, desde que haja uma grande concentração de áreas comerciais ou administrativas. Além disso, conforme a estrutura viária na qual o estacionamento está, ele pode ser o causador de congestionamentos, pelo fato de bloqueio de tráfego durante a saída ou entrada em vagas. (LUZ, 1997).

Como citado anteriormente, Luz (1997) escreve os problemas com estacionamentos são cada vez mais notáveis e que apesar do aumento das vagas, ainda assim não será possível atender à demanda das cidades. Além disso, um grande fator que influencia isso são os investimentos muito superiores em estrutura viária do que na própria estrutura de estacionamentos.

Adicionalmente, com o aumento no volume de vendas de carros o problema de falta de estacionamento também se agravada, ainda mais se ligado à cidades onde há um sistema de transporte público precário. Para conseguir melhorar a situação e reduzir o esgotamento de vagas em estacionamento muitos fatores deveriam contribuir, entre eles um melhor acesso ao transporte público (SILVA, 2011).

Em Silva (2011), o autor escreve uma breve classificação das categorias de estacionamentos, quanto à sua utilização. Os estacionamentos podem ser classificados como:

- públicos, os quais qualquer pessoa pode utilizar;
- privados, em que apenas pessoas autorizadas têm acesso, como estacionamentos de empresas por exemplo;
- pagos ou rotativos, nos quais o motorista paga para utilizar o espaço por algum tempo. Este que pode ser considerado como uma ramificação do estacionamento público, desde que a pessoa pague para usar.

Normalmente ao buscar vagas, o motorista apresenta um padrão de ações principais, entre elas encontram-se (SILVA, 2011):

1. Entrada no estacionamento, ou na rua onde se busca estacionar;
2. Busca pela melhor vaga disponível;
3. Entrada na vaga e estacionamento do veículo, ocupando a vaga;
4. Saída do estacionamento deixando a vaga livre novamente.

Além disso, durante a escolha da vaga de estacionamento o motorista toma várias decisões, uma delas em especial é a distância ao local de destino. As pessoas tendem a estacionar o mais próximo ao seu destino final, para evitar caminhadas longas. Quanto maior a cidade, maior tende a ser distância caminhada entre o lugar onde o carro está estacionado e o destino final (LUZ, 1997).

Desta forma, com a informação de um local com melhor probabilidade de ser encontrado um lugar para se estacionar e que esteja perto do seu destino, o motorista pode tomar essa decisão mais rapidamente e evitar a circulação desnecessárias pelos estacionamentos em busca de uma vaga.

A próxima seção descreve características e funcionamento do GPS, ferramenta a ser utilizada pelo trabalho para obter os dados de localização dos veículos.

2.2 *Global Positioning System (GPS)*

Nesta seção será descrito o funcionamento do *Global Positioning System (GPS)*, porém, primeiramente serão descritos alguns termos relacionados à geografia e que fazem parte dos dados que são usados no GPS, principalmente referente à coordenadas geográficas. Porém, para iniciar da definição do GPS é necessário entender alguns conceitos que estão ligado ao posicionamento geográfico, principalmente quando o assunto está ligado à latitude e longitude, que são os dados retornados pelo GPS.

A definição de posição é um conjunto de coordenadas em um sistema de coordenadas definido. A forma mais comum de expressar uma posição geográfica terrestre é utilizando os ângulos de latitude e longitude. Os dois ângulos definem um ponto na superfície elipsoide do globo terrestre (OS, 2015).

As linhas que cruzam de norte a sul são chamadas de meridianos e representam a latitude, já as linhas que cruzam de leste a oeste são chamadas de paralelas, e estas representam a longitude (OS, 2015). Para a latitude é medida usando uma variação positiva de 0° a 90° *North*, ou então com uma variação negativa de 0° a 90° *South*. A Longitude, pelo contrário varia de 0° a 180° *West* ou de 0° a 180° *East*. Quando se está em no meridiano 0° , tem-se o chamado Meridiano de Greenwich, assim como quando se está sobre o paralelo 0° tem-se a Linha do Equador (IBGE, 1987).

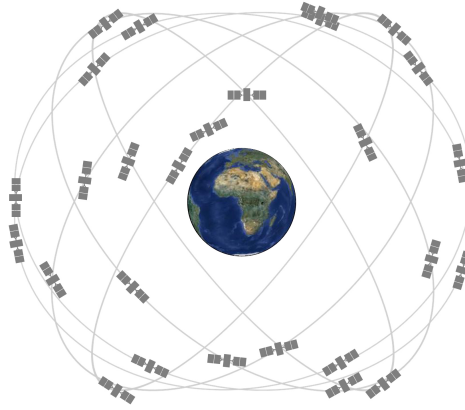
A partir das definições de posição descritas acima, é possível fazer a definição sobre o que é e o como é o funcionamento do GPS. O sistema foi criado e desenvolvido por volta de 1960 pelos Estados Unidos da América (EUA) para realizar o monitoramento dos seus submarinos que carregavam mísseis nucleares durante a Guerra Fria (NASA, 2014). Por meados de 1993 essa ferramenta foi liberada para uso civil pois o governo norte americano percebeu que muitas aplicações poderiam utilizar esse serviço para beneficiar a população (NCO, 1993)

Entretanto, apesar do uso do GPS ser liberado para a população, a parte liberada para uso civil não é tão precisa quanto a que continua sendo utilizada com intuítos militares. Como escrito na norma técnica pelo NCO (1993), em sua primeira versão, o sistema habilitado para uso civil é chamado de *GPS Standart Positioning Service* (SPS). Esta versão do GPS possui uma precisão muito inferior à utilizada em operações militares que é o chamado *GPS Precision Positioning Service* (PPS). Por esse motivo é possível entender o porquê acontecem muitas vezes um erro de precisão na localização ao utilizar o GPS de *smartphones*, por exemplo, sua precisão possui uma margem de erro.

Com o passar dos anos, o sinal de GPS SPS foi melhorado, e cada vez mais sua margem de erro tem diminuído, porém sua precisão ainda continua inferior à do GPS PPS que possui um número maior de satélites e estes são mais específicos para captura da posição. Para título de curiosidade, todas as normas técnicas do GPS, desde sua liberação para o uso civil, encontram-se na página do governo americano sobre o GPS, que pode ser encontrada em NCO (2016). A *National Aeronautics and Space Administration* (NASA) também possui uma página com informações sobre GPS, que está disponível em NASA (2014).

Realizando um resumo de como o GPS SPS funciona, ele possui vinte e quatro satélites que se estão espalhados em seis órbitas do Planeta Terra e que orbitam a Terra duas vezes por dia a aproximadamente 20.200 km de altura, como é possível de se visualizar na Figura 1.

Figura 1 – Constelação de satélites de GPS na órbita do Planeta Terra



Fonte: NASA (2014).

A constelação dos satélites apresentada na Figura 1 consiste no chamado Segmento Espacial do GPS, que possui também o Segmento de Controle, que realiza o monitoramento e a manutenção dos satélites na Terra, gerenciados pelo governo dos EUA, e o Segmento de Usuários, onde se encaixam qualquer aparelho GPS que processa os dados de navegação e calculam a posição (Latitude e Longitude) e o tempo (NASA, 2014).

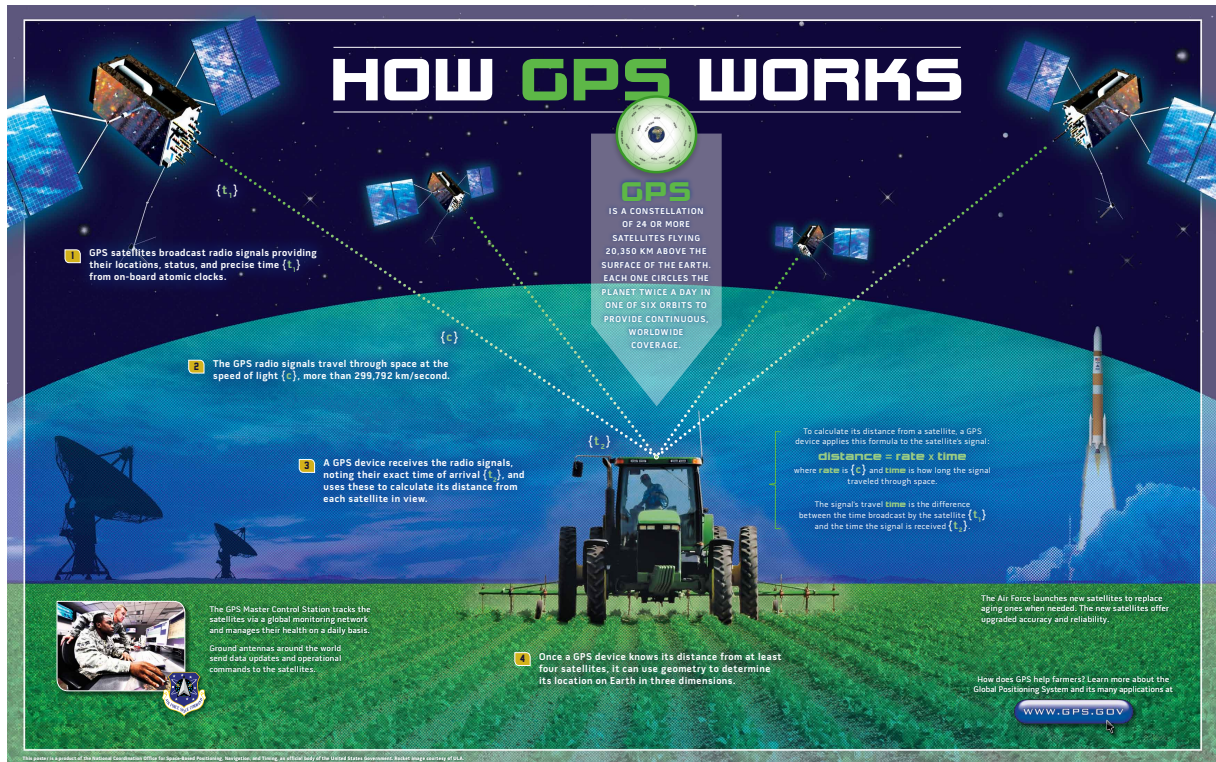
A Figura 2 mostra o funcionamento do GPS, cada um dos pontos amarelos da figura está descrito abaixo nos passos de como o aparelho recebe o sinal e faz o processamento para receber sua localização:

1. Os satélites de GPS fazem o *broadcast* de sinais de rádio com a sua localização, status e o tempo real (t_1) baseado em relógios atômicos que estão presentes dentro deles;
2. O sinal de GPS viaja através do espaço na velocidade da luz (c), em aproximadamente 299.792 km/s ;
3. Quando o dispositivo de GPS recebe o sinal, ele utiliza o tempo em que o sinal chegou (t_2) e para calcular a distância (s) entre o dispositivo e o satélite de quem recebeu o sinal, utilizando o cálculo abaixo, que é utilizado para encontrar a distância de cada todos os satélites dos quais o dispositivo recebe algum sinal.

$$s = c(t_2 - t_1)$$

4. Assim que o dispositivo de GPS conhece sua distância de pelo menos quatro satélites, ele então pode utilizar geometria para encontrar sua localização na Terra em três dimensões (Latitude, Longitude e Altitude).

Figura 2 – Funcionamento do sistema de GPS



Fonte: NCO (2013).

Como pode ser visto, há um grande sincronismo entre os satélites e o dispositivo para realizar a comunicação e determinação da posição. Este pode ser um problema, pois pode acontecer algum atraso nas mensagens ou mesmo a perda sinal de um dos satélites, sendo necessário utilizar um outro para realizar o cálculo de posição. Dessa forma alguns problemas, como a imprecisão do sinal de GPS em determinados locais, principalmente em locais fechados, e dispositivos móveis com diferentes qualidades, explicam as possíveis falhas na comunicação e que podem levar a um diferente comportamento da aplicação.

A seção seguinte apresenta o termo velocidade, que mede o quanto houve de deslocamento durante um período de tempo, e é um dos valores que serão utilizados para obter-se o estado do veículo.

2.3 Velocidade

A velocidade é uma grandeza que mede o quanto houve de deslocamento durante um período de tempo, por exemplo, quantos quilômetros foram percorridos durante quantas horas, e ela é medida em km/h ou m/s , dependendo da abrangência ou o que deseja-se obter. Para explicar de uma forma mais detalhada, é necessário dividir a velocidade em duas partes. A parte de velocidade média e a parte de velocidade instantânea, que serão apresentadas a seguir.

2.3.1 Velocidade Média

A velocidade média é a divisão do deslocamento pelo intervalo de tempo em um certo intervalo de tempo. Dessa forma a velocidade média é a função de dois instantes de tempo, $V_m(t_1, t_2)$, sendo o deslocamento a diferença da distância entre dois pontos, $s_2 - s_1$ (PALANDI et al., 2010).

A velocidade média, resumidamente, é o valor da rapidez com que um corpo se move entre duas posições, em linha reta, não afirmando que o corpo manteve esta velocidade sempre, pois ele pode ter tido alguma mudança de velocidade no meio do caminho, o que simplesmente ela informa é que a pessoa se deslocou s distância em t tempo (VEIT, 2007). Desta forma pode-se afirmar que a equação para determinar a velocidade média é:

$$V_m = \frac{\Delta s}{\Delta t}$$

Sendo Δs a variação de distância entre s_2 e s_1 e Δt a variação de tempo entre t_2 e t_1 .

Porém, a velocidade média sempre está ligada a dois instantes de tempo, por este motivo chamado de média. No entanto, a velocidade que é mais utilizada para expressar a velocidade de qualquer objeto é a velocidade instantânea, que é a velocidade em um instante de tempo específico (PALANDI et al., 2010).

2.3.2 Velocidade Instantânea

Como citado na subseção anterior, a velocidade média é calculada entre dois instantes de tempo, porém a velocidade média não é uma medida muito utilizada para expressar a velocidade, mas sim a velocidade instantânea, que é o deslocamento em um instante específico de tempo, ou seja em praticamente zero.

Para definir o que é velocidade instantânea, é preciso utilizar um conceito da matemática,

que é chamado de limite. Como na velocidade média se utilizam dois tempos para calcular a variação de tempo, para poder obter uma variação de tempo que seja um determinado instante, é necessário ter um intervalo cada vez menor, ou seja que praticamente não haja diferença entre t_1 e t_2 , para isso se usa o limite que tende a zero, de um só instante de tempo (MARQUES, 2008). Desta forma a equação de cálculo de velocidade pode ser escrita da forma abaixo:

$$V_i = \lim_{\Delta t \rightarrow 0} \frac{\Delta s}{\Delta t}$$

Em uma definição mais básica de velocidade instantânea, ou simplesmente velocidade, pode-se afirmar que é a diferença se dá no fato de que a variação de tempo é utilizada como sendo o infinitamente zero, dessa forma o valor final da equação se torna a velocidade daquele exato instante (VEIT, 2007).

Para a solução a ser propostas, ambas as velocidades citadas acima serão utilizadas, desde a média quanto a instantânea, visto que para uma análise de obter o status de um objeto em movimento é necessário verificar sua velocidade média, mas da mesma forma a velocidade em um momento específico visando um histórico de velocidade máxima.

Uma melhor definição de o porquê serão utilizados esses dados e o porquê da divisão em duas medidas de velocidade é o fato de que para se obter o estado de veículo parado é necessário analisar a velocidade daquele momento em que ele parou, ou seja, a velocidade instantânea.

Já a velocidade média não é uma medida que ajuda a definir se um veículo está parado ou não, pois ela leva em conta todo o período e a distância total, e apenas chega a zero depois de muito tempo. Assim que um carro para, por exemplo, seu valor começa a cair pois não há mais variação, mas não chegará a zero instantaneamente como é com a velocidade instantânea, que leva em conta apenas um período quase nulo. Porém, ela será utilizada da mesma forma para se obter um histórico, e poder utilizar o mesmo para ajudar na determinação de um padrão para determinar em qual o tipo de veículo a pessoa está.

Tendo em base que não se possui um movimento uniforme, será necessário além da velocidade utilizar mais um dado que se extrai do movimento, que é a aceleração de um objeto. Saindo do repouso para um movimento, é a aceleração é a responsável pela variação da velocidade, assim como quando o objeto está em movimento e aumenta sua velocidade. Na próxima seção será apresentado o termo aceleração juntamente com cálculos para o melhor entendimento do que é e de como se obtém este valor.

2.4 Aceleração

A aceleração é o que faz um objeto sair do repouso e iniciar o movimento e por isso é uma medida definida a partir da velocidade do objeto. Sua definição básica é o quanto a velocidade de um corpo variou em um intervalo de tempo. Por exemplo, se um objeto ao iniciar o movimento possui uma aceleração de 10km/h/s , isso indica que a cada segundo o objeto irá aumentar sua velocidade em 10km/h , seguindo o exemplo, em 2s sua velocidade seria de 20km/h . Assim, a aceleração se define pela divisão da variação da velocidade pelo intervalo de tempo correspondente. A aceleração é medida em m/s/s ou simplesmente por m/s^2 (VEIT, 2007).

Se por eventual um objeto estiver em velocidade constante, a sua aceleração será nula, visto que não há variação de velocidade. Definindo formalmente a aceleração de um objeto, é a divisão da variação de velocidade por uma variação de tempo, e é calculada pela equação descrita abaixo, sendo Δv como a variação da velocidade e Δt como a variação de tempo (MARQUES, 2008):

$$a = \frac{\Delta v}{\Delta t}$$

Porém, assim como na velocidade descrita na seção anterior, a aceleração que é de mais importância é a aceleração instantânea e não a média. Ou seja, a aceleração entre uma mínima variação de tempo, em que Δt se aproxime o máximo de zero. Desta forma a aceleração torna-se praticamente a aceleração naquele instante (VEIT, 2007).

Para realizar o cálculo é necessário utilizar intervalos de tempo cada vez menores, ou seja, buscando o limite em que a variação de tempo se aproxime de zero, em que t_2 seja muito próximo de t_1 . Assim, a aceleração instantânea pode ser calculada utilizando limites, assim como a velocidade, e sua equação pode ser descrita por (MARQUES, 2008):

$$a = \lim_{\Delta t \rightarrow 0} \frac{\Delta v}{\Delta t}$$

Assim como escrito acima sobre ambas as velocidades serem utilizadas na solução a ser proposta, o mesmo se aplica para a aceleração, pois a aceleração também será um parâmetro utilizado para determinar o estado do veículo, e muito usada para obter qual o tipo do veículo, visto que cada meio de locomoção possui acelerações diferentes, desde acelerações médias entre um trecho e aceleração instantânea de um determinado instante.

Da mesma forma apresentada na seção sobre velocidade, a aceleração instantânea se torna mais importante para determinar o estado final de um veículo, e a média servirá para um histórico de mudança de acelerações durante uma viagem, ajudando, juntamente com os dados de velocidade média a determinar o tipo de veículo no qual a pessoa está.

Como o trabalho não possui o objetivo de realizar o cálculo destes dados, pois eles são automaticamente feitos pelo GPS e utilizando o próprio dispositivo através do acelerômetro, não pretende-se fazer o aprofundamento dos cálculos para a obtenção dos valores.

Neste capítulo foram apresentados os principais termos teóricos que envolvem o trabalho. Termos estes que possuem grande utilização em pesquisas relacionadas a deslocamentos e determinação de estados, principalmente que envolvam o uso de GPS. No próximo Capítulo serão descritos e realizados comparativos entre os trabalhos relacionados, que utilizam alguns testes termos apresentados, e que buscam embasar a solução proposta por este, além da comparação dos mesmos com o presente trabalho.

3 TRABALHOS RELACIONADOS

Neste capítulo, serão citados e comentados trabalhos relacionados ao presente trabalho proposto com intuito de apresentar referências sobre o assunto e como está o estado da arte na pesquisa sobre monitoramento de movimentos que envolvam GPS e tecnologias semelhantes, e que possam ajudar no processo detecção de estado de movimento e informar se o objeto está parado ou ainda em movimento.

O trabalho de **Ohashi et al. (2016)** tem como principal objetivo a criação de um método para extrair com precisão os períodos de deslocamento, descritos no artigo como *trip periods*, chamado de *trip-extraction method* que utiliza a origem, o destino, o tempo de saída e chegada para efetuar o cálculo.

Os autores citam exemplos e referências que utilizam diferentes métodos de extração e de cálculos para encontrar os períodos de paradas e de deslocamento. A ideia das referências citadas no artigo é de uma corrigir e aprimorar o que havia sido feito na outra. Inclusive, neste artigo, os autores citam um próprio trabalho antigo onde utilizaram um outro método para o cálculo, que ajustava dinamicamente a forma e o tamanho da borda para detectar área de paradas de acordo com a distribuição dos dados de GPS coletados.

O problema do antigo trabalho do autor, era de detectar pequenas distâncias pois a borda também aumentava, além disso descrevem que o uso de GPS pode afetar a extração de dados, gerando falsos positivos por causa de erros de posição e problemas para detectar deslocamentos pequenos. Dessa forma, o presente trabalho foi apresentado para buscar aprimorar e corrigir o problema descrito acima.

A proposta dos autores é de que cada ponto GPS seja classificado como ponto de parada (*stay point*) ou ponto de deslocamento (*trip point*). E a partir deles, buscar encontrar um método para detectar automaticamente os períodos de deslocamentos ou de parada, baseados em coletar dados de GPS continuamente. Para realizar os cálculos cinco variáveis foram usadas:

1. Precisão do GPS;
2. Diferença de tempo entre o ponto atual, o anterior e o próximo;
3. Semelhança entre direções;
4. Velocidade e;
5. Mudança de aceleração.

Todos estes dados foram calculados pra cada dado recebido do GPS. Sendo que os dados de

precisão do GPS e velocidade foram obtidos utilizando o *Google Maps* (GOOGLE DEVELOPERS, 2016e), e os outros dados calculados em equações baseadas nos dados obtidos através do GPS.

Após analisar os resultados, Ohashi et al. (2016) chegaram à conclusão que o método proposto teve uma precisão bem maior que os anteriores citados e conseguiu detectar vários pontos de parada entre um deslocamento e outro. Além disso, verificaram que a velocidade e aceleração não contribuíram tanto quanto mudança de direção, diferença de tempo e posição de GPS no cálculo da precisão final.

Já o trabalho de **Huang et al. (2016)**, se baseia na análise do comportamento de viagens e deslocamentos de usuários e desta forma buscar encontrar os chamados *stop points*, áreas onde há, por exemplo, uma pausa em uma viagem, tudo baseado nos dados de GPS. O trabalho propõe um método chamado de *Mining Repeat Travel Behaviors Using Stop Regions* (MRTBUSR) cuja ideia principal é encontrar estas regiões e as agrupar em *clusters*.

Após o agrupamento os autores buscam encontrar sequências semelhantes de deslocamento nestas regiões para criar rotas que tenham possível repetição, analisando o tempo de duração de um deslocamento, a posição, entre outros fatores, com o objetivo de encontrar algum comportamento de deslocamento do usuário.

Huang et al. (2016) calcula o ponto de estabilidade dividindo a variação de tempo, entre o tempo final e inicial, pelo comprimento do trajeto entre as posições B e A. A partir disso, é feito um corte, em que os pontos que tiverem valor maior que o *threshold* definido são mantidos, e o resto descartado. Estes que são mantidos são utilizados como os *stop points* que podem ser agrupados e a partir deles serem aplicados os cálculos de mineração de dados para encontrar um comportamento de viagens e deslocamentos.

Algumas referências citadas pelos autores em Huang et al. (2016) usam técnicas mais simples. Os autores citam próprios trabalhos que ajudaram a escrever anteriormente e que também utilizam GPS, tempo e inteligência artificial para determinar comportamentos, em que o mais recente, que foi chamado de *Duration of Staying on each location of Trajectories* (DoS-Tra), consegue identificar comportamentos usando o tempo de duração em que a pessoa ficou em determinado local para encontrar grupos com estilos de vida similares (GUO et al., 2015).

No artigo escrito por **Wang et al. (2015)**, os autores exploram a previsibilidade de um tráfego urbano e o congestionamento se baseando em dados de GPS instalados em táxis que enviam dados em tempo real, os dados são a velocidade e a localização. Com estes valores é

possível investigar e qualificar a previsão das principais estradas e ruas. No trabalho, os autores estabelecem um mapa com o grau de congestionamento em um segmento da rua em uma série de símbolos que permite a eles aplicar ferramentas e calcular medidas que possam ajudar a prever a condição do tráfego naquele segmento da estrada ou rua avaliada.

A pesquisa foi realizada na cidade de Pequim, na China. Para realizar a pesquisa, os autores dividiram a cidade em três anéis, onde cada anel representava uma parte da cidade. Começando com o segundo anel (2) localizado mais no centro, o terceiro (3) na área mais central e o quarto anel (4) localizado na parte mais distante do centro da cidade. Após a análise dos dados, os autores chegaram à conclusão de que nos anéis mais centrais a velocidade tende a ser menor que nos externos, justamente pelo maior tráfego.

Mesmo sem conhecer nenhum padrão sobre os motoristas, como origem e destino, o uso do mapeamento pode encontrar relativamente um alto índice de previsão de condições de tráfego nos anéis. Além disso uma informação interessante obtida foi de que como o terceiro anel se encontra entre o anel com menor velocidade média e anel o que possui maior velocidade média, o terceiro anel foi o mais difícil de se prever a condição do tráfego e sua velocidade média, justamente pelo fato de haver uma mistura de dois tráfegos diferentes.

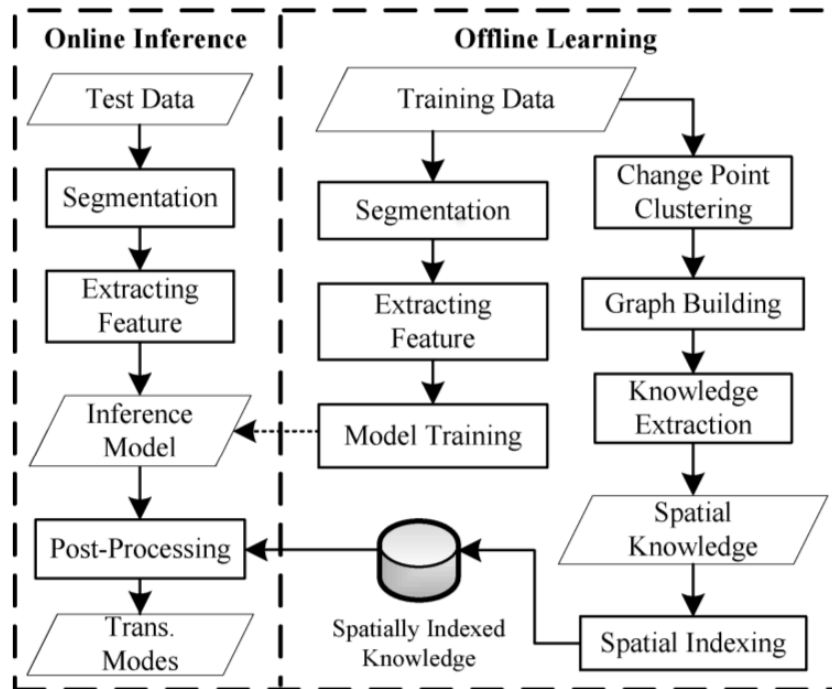
Como conclusão os autores detectaram que o uso do GPS nos veículos para capturar os padrões de uma rua juntamente com os cálculos de entropia e a inequação de Fano demonstraram que os padrões diários de tráfego nos três anéis principais de Pequim foram bem previsíveis, ao menos baseados em um histórico de curto tempo e sem nenhum conhecimento dos padrões e nem dos comportamentos dos motoristas.

Em **Zheng et al. (2010)** os autores procuram compreender o comportamento dos meios de transporte em áreas urbanas, utilizando dados de GPS comparando funcionalidades básicas de GPS com funcionalidades avançadas para diferenciar os meios de locomoção, dividindo os movimentos entre segmentos de *walking* (caminhada) e *non-walking* (não caminhada), ao invés de dividir entre diferentes tipos de meios de transporte, como carros, ônibus e bicicletas.

A arquitetura do trabalhos foi dividida em duas partes, a parte *offline* e a parte *online*, como é apresentado na Figura 3, e dentro de cada uma determinadas ações foram tomadas. A partir de todos os dados coletados por aparelhos de GPS que foram entregues a vários usuários durante dez meses, foi salvo um grande *log* de informações que foram utilizados, uma parte para treinar o programa e outra parte para que serviu como entrada dos dados.

Na parte *offline*, uma parte dos dados de GPS colhidos foram utilizados como *training*

Figura 3 – Arquitetura do trabalho de Zheng et al. (2010)



Fonte: Zheng et al. (2010).

set. Após entrar em eles no sistema, os dados foram segmentados, baseados em pontos de troca, que são caracterizados como momento em que a pessoa troca de um veículo para caminha a pé, ou para outro veículo. Esses dados segmentados são então utilizados para treinar um modelo de classificação, que depois é separado em *clusters* para agrupar os pontos de troca e criar um grafo baseado nos grafos, gerando trajetórias para cada usuário.

Após, para a parte *online*, o que sobrou dos dados foi usada para ser realmente testada e para ver determinar se a arquitetura funcionaria e identificaria o tipo de transporte usado. Ao entrar com os dados eles eram segmentados e suas informações lidas, como velocidade, aceleração, distância, variações de paradas e mudança de direção.

Posteriormente, essas informações eram comparadas com o modelo de treinamento, e analisadas junto ao grafo criado na parte de treinamento. Feito isso, o resultado dos dados de entrada eram obtidos.

Como conclusão, verificaram que o uso de dados mais avançados, como as variações de velocidade, direção e aceleração, foram mais importantes para detectar qual o modelo de transporte utilizado por uma pessoa após uma parada ou troca de veículo, se comparadas com apenas a utilização dos dados fixos como velocidade e aceleração.

Além disso, apesar de alguns falsos positivos, o trabalho pode detectar com um acerto por

volta 70% a diferenciação dos meios de locomoção, baseado nas mudanças de comportamento das pessoas, que trocavam de veículo, ou que iniciavam um segmento de caminhada e logo depois utilizavam um veículo.

Com objetivo de finalizar o capítulo de trabalhos relacionados, é apresentada uma tabela comparativa entre os trabalhos descritos acima, apresentando brevemente a abordagem de cada um deles.

Tabela 1 – Comparativo de trabalhos relacionados

Trabalho	Objetivo	Estratégia	Dados utilizados	Tecnologia
Ohashi et al. (2016)	Extrair períodos de deslocamento dentro de uma viagem	Cada ponto de uma viagem é considerado como de parada ou deslocamento, e a partir deles são extraídos os períodos de cada um	Uso de informações de geolocalização, direção, velocidade, tempo e aceleração	GPS
Huang et al. (2016)	Encontrar comportamentos de deslocamento dos usuários	Utilizar <i>machine learning</i> para agrupar os deslocamentos, obtidos através dos dados capturados pelo GPS, em <i>clusters</i>	Uso de dados de GPS e variação de tempo	GPS e <i>machine learning</i>
Wang et al. (2015)	Previsibilidade de tráfego urbano e congestionamento	Pequim foi dividida em 3 anéis, e cada um analisado com objetivo de detectar a situação de tráfego em cada um dos anéis	Dados de localização de táxis dentro da cidade e velocidade dos mesmos	GPS
Zheng et al. (2010)	Compreender o comportamento dos meio de transporte em um cenário urbano	Utilizar funcionalidades básicas de GPS e avançadas para treinar um modelo, e utilizá-lo para comparar com dados reais para diferenciar meios de locomoção	Localização, distância, tempo e variações de velocidade, de paradas e de mudança de direção	GPS e <i>machine learning</i>
Culau (2016)	Definir se um veículo está estacionado e obter mapa com preenchimento dos estacionamentos	Obter o estado de veículos para determinar se estão estacionados e utilizar esta informação para determinar o preenchimento de estacionamentos	Uso de dados de localização, distância, velocidade, aceleração e variação nos sensores do dispositivo para determinar se veículo está estacionado	GPS, aceleração e sensores do dispositivo móvel

Fonte: Elaborado pelo autor.

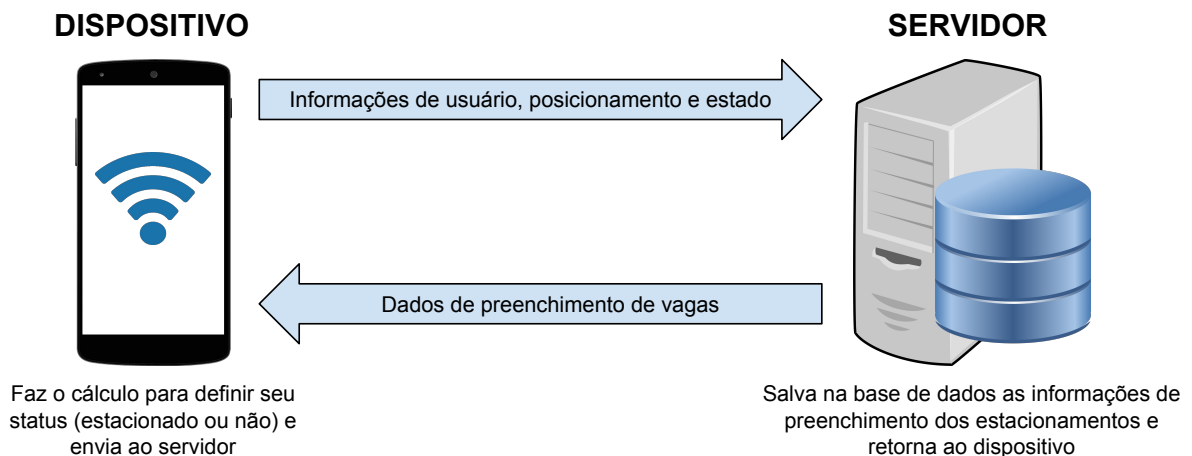
4 SOLUÇÃO DESENVOLVIDA

Este capítulo descreve como foi o desenvolvimento do trabalho, juntamente com a definição do que foi utilizado para o desenvolvimento, além de como foram feitos os testes.

Inicialmente, fazendo referência à proposta inicial do trabalho, o objetivo do aplicativo é de definir o estado de um veículo, obtendo a informação de se ele está ou não estacionado, e, a partir delas, avisar o usuário sobre o preenchimento de determinadas áreas de estacionamentos, ajudando o mesmo a decidir o lugar onde ele tenha mais possibilidade de encontrar vagas disponíveis.

A solução proposta para o trabalho foi focada em planejar e definir uma solução que pudesse encontrar um padrão para a definição do *status* de um veículo (estacionado ou não estacionado), a partir dos dados de sensores do dispositivo. Além disso, foi definido que os dados seriam armazenados em um servidor e que o aplicativo faria a consulta e atualização dos dados, enviando os dados necessários para o servidor, e este, retornaria os valores a serem exibidos para o usuário em um mapa. A Figura 4 exibe o fluxo básico da solução proposta como descrito anteriormente.

Figura 4 – Fluxo de dados entre dispositivo e servidor



Fonte: Elaborado pelo autor.

Nas seções seguintes, será descrito o desenvolvimento do trabalho, exibindo como foram definidos os fluxos de dados, sua obtenção, a detecção de estado do aplicativo e a comunicação entre as partes (dispositivo móvel e servidor), e por fim uma seção com os testes realizados e dificuldades encontradas durante o desenvolvimento.

4.1 Desenvolvimento

O desenvolvimento da aplicação para dispositivos móveis foi feita apenas para o sistema operacional *Android* (GOOGLE DEVELOPERS, 2016b), utilizando o *Google Maps* (GOOGLE DEVELOPERS, 2016e) como ferramenta para mapas. A captura de dados do dispositivo foi realizada diretamente pela Application Programming Interface (API) de desenvolvimento, utilizando funções de localização e detecção de movimento. A versão do sistema operacional *Android* utilizada foi a versão 6 (*Android Marshmallow*) com API versão 23 (GOOGLE DEVELOPERS, 2016b).

O dispositivo utilizado para testes foi um *smartphone Nexus 5* (GSM ARENA, 2015), desenvolvido pela *Google* em parceria com a *LG* e lançado em outubro de 2013. Apesar de já ter dois anos de idade, o celular ainda conta com atualizações recentes e possui um *hardware* semelhante a muitos *smartphones* lançados atualmente, o que o torna um aparelho competente e confiável.

A versão do aplicativo abrange desde a versão 4.1 (*Jelly Bean*) até a versão 6 (*Marshmallow*), que são as que possuem a maior fatia de uso no mercado, como é mostrado na Figura 5. Recentemente, a *Google* fez o lançamento da versão 7 (*Nougat*), porém foi lançada apenas para poucos dispositivos, porém o aplicativo funciona tranquilamente pelo fato de que as APIs seguem o padrão de que versões mais novas, no caso da API versão 24 do *Android 7 (Nougat)*, utilize as funcionalidades das mais antigas.

Esta Seção foi separada em duas partes, aplicativo e servidor, abordando separadamente cada uma das partes do desenvolvimento, com objetivo de facilitar a compreensão da ferramenta desenvolvida, com um maior detalhamento e exemplos de fluxo de dados e estruturas de dados utilizadas.

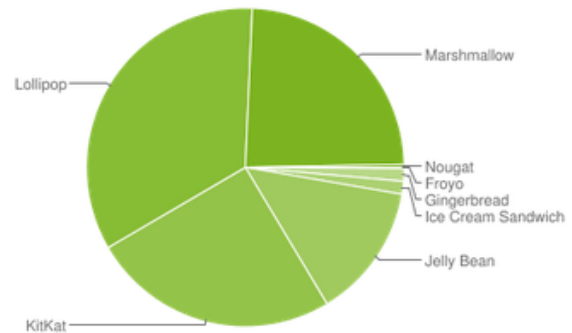
4.1.1 Estacione UNISC

O aplicativo *Estacione UNISC* é a parte principal do trabalho, pois é dentro dele onde todo o processamento é feito, desde a detecção da localização e detecção da atividade que o usuário está executando, até a comunicação com o servidor e exibição do mapa para o usuário final com os dados de preenchimento dos estacionamentos.

Desde o início do seu desenvolvimento, o aplicativo teve o principal objetivo de fazer a detecção de como o usuário está se movendo, buscando sempre detectar o estado de estar em movimento em um veículo, e além disso, obter a localização a partir do GPS e sua velocidade. O

Figura 5 – Gráfico com a distribuição de uso de cada versão do Android

Version	Codename	API	Distribution
2.2	Froyo	8	0.1%
2.3.3 - 2.3.7	Gingerbread	10	1.3%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	1.3%
4.1.x	Jelly Bean	16	4.9%
4.2.x		17	6.8%
4.3		18	2.0%
4.4	KitKat	19	25.2%
5.0	Lollipop	21	11.3%
5.1		22	22.8%
6.0	Marshmallow	23	24.0%
7.0	Nougat	24	0.3%



Data collected during a 7-day period ending on November 7, 2016.

Any versions with less than 0.1% distribution are not shown.

Fonte: (GOOGLE DEVELOPERS, 2016c)

desenvolvimento ocorreu em partes separadas, justamente para poder facilitar a implementação de cada funcionalidade.

Inicialmente, foi necessário estudar o desenvolvimento de aplicativos em *Android*, pelo fato de que nunca havia-se trabalhado com esta plataforma anteriormente. A plataforma de desenvolvimento é baseada em *XML* para a parte de interfaces, *Java* para o código em si, para base de dados interna utiliza o banco de dados *SQLite*, e além disso possui algumas particularidades para transferir dados entre classes e entre telas.

Através de leituras e exemplos, foi iniciado o desenvolvimento da primeira tarefa que foi de obter a localização e a velocidade na qual o dispositivo móvel se encontrava em um determinado momento. E, a partir destes dados, definir como seria feito o cálculo para a obter o seu estado, assim como definir o formato dos dados a serem enviados para o servidor.

4.1.1.1 Localização e velocidade

A primeira tarefa desenvolvida foi a de buscar as informações relacionadas à geolocalização e partir dela, obter a velocidade do dispositivo. Após os estudos citados anteriormente, foi

desenvolvido um protótipo, no qual era apenas uma tela com campos de texto que atualizavam o valor assim que a localização se modificasse. E, a partir da localização, foi possível obter a velocidade em que o dispositivo se deslocava.

Para o desenvolvimento desta parte, foi utilizado a API de desenvolvimento do *Android*, já citada anteriormente, que possui uma biblioteca, chamada *Location*, que faz todo o tratamento de consultas à localizações, lendo os dados do sensor de GPS presente no dispositivo. Desta forma não foi necessário trabalhar com o *hardware* do dispositivo para ler os dados posição geográfica, valor que é apresentado como um valor *double* com valores de latitude e longitude.

Juntamente com a consulta dos dados de GPS, esta biblioteca, também possui um cálculo de velocidade em *m/s*, em que calcula, por pequenos períodos de tempo, a variação da posição geográfica e desta forma obtém o valor para a velocidade deste deslocamento, que para ser convertida para *km/h*, é necessário multiplicar pelo número 3,6.

Com a parte de localização e velocidade desenvolvida, foi necessário iniciar a parte de detecção do estado de como o usuário está se comportando, se está parado ou em movimento, e se estiver em movimento, como está o fazendo, se através um veículo, caminhando, correndo ou andando de bicicleta, por exemplo. Na próxima subseção será feita a descrição de como foi o desenvolvimento desta parte do trabalho.

4.1.1.2 Detecção do tipo de movimento do usuário

A detecção do tipo de movimento do usuário, desde o início tinha por estratégia utilizar a aceleração do dispositivo, buscando este dado junto ao acelerômetro do mesmo. Durante a busca por informações relacionadas, e formas de calcular, foi encontrado em exemplos e tutoriais, que a aceleração provinda do acelerômetro provém da variação da gravidade aplicada ao dispositivo. Desta forma qualquer vibração aplicada ao mesmo, era detectada como sendo aceleração, e o que era desejável era obter apenas a aceleração em caso de movimentação para frente ou para trás, e não de qualquer pequena vibração.

Como não se possuía nenhum conhecimento nesta parte, algumas tentativas sem sucesso foram feitas, principalmente com a aplicação de filtros para remover variações em eixos específicos, porém todas com resultados negativos. Após algum tempo pesquisando maneiras de encontrar a aceleração do dispositivo, foi encontrado um material no site de desenvolvedores da *Google*, onde justamente era apresentada uma API que consegue obter estes dados e analisá-los de forma a obter a possível atividade que o usuário estava realizando.

Esta API do *Android* é chamada de *ActivityRecognitionAPI* (GOOGLE DEVELOPERS, 2016a) (GOOGLE DEVELOPERS, 2015), que consegue detectar a possível atividade que o usuário está realizando e retorná-la com um grau de confiabilidade. A detecção é feita através da leitura periódica dos dados de todos os sensores do aparelho (localização, velocidade, giroscópio, acelerômetro, e outros) e pela aplicação de um algoritmo de *machine learning*, utilizando modelos já existentes internamente na API, e fazendo com que durante o momento de detecção seja possível obter estes dados de possível atividade que está sendo executada.

O seu funcionamento detalhado se inicia com a criação do objeto e a chamada do método mostrado no trecho de código Código 4.1. Esta chamada faz com que o aplicativo chame os métodos da API de desenvolvimento do *Android*, através da variável *mGoogleApiClient*, e esta faz a leitura toda na camada mais inferior do sistema operacional. Desta forma, não afeta o desempenho do aplicativo, apenas envia gatilhos, conforme o tempo configurado em *DETECTION_INTERVAL_IN_MILLISECONDS*, para um *listener*, o chamado *pendingIntent*, que fica ouvindo se não há nada novo. Assim que receber algo da API, envia de volta para o aplicativo e então este dado é tratado e aplicado nos tratamentos posteriores.

Código 4.1 – Chamada para a *ActivityDetectionAPI* atualizar os dados

```
ActivityRecognition.ActivityRecognitionApi.requestActivityUpdates (
    mGoogleApiClient,
    Constants.DETECTION_INTERVAL_IN_MILLISECONDS,
    pendingIntent
);
```

A grande vantagem da utilização desta API é o fato de que tudo é feito independente do aplicativo, rodando em *background*. Os dados são processados diretamente em nível de sistema operacional, em que os sensores são lidos, e os retornos de cada um aplicados a um algoritmo de aprendizado de máquina, onde são analisados e verificados para ver com qual dos tipos de padrões ele mais se parece, e com qual a confiabilidade desta análise. Assim que obtidos, são então enviados ao *listener*, que como descrito anteriormente devolve para a aplicação. O *app* simplesmente colhe os dados já formatados sem ser necessário ler na camada mais alta, desta forma é possível obter dados em tempo de execução, caso contrário poderia-se ter atraso no processamento dos dados, se tudo fosse lido e processado diretamente no aplicativo.

Esta detecção feita pela API de reconhecimento de atividades, varia conforme o passar do tempo, sendo cada vez mais precisa enquanto faz a análise, como citado nas referências desta API, a mesma fica analisando no *background*, porém no aplicativo desenvolvido as atualizações de estado de estacionado para não estacionado, e vice-versa, apenas são feitas se o aplicativo estiver aberto ou rodando minimizado. Dessa forma, seu funcionamento é semelhante aos aplicativos de trânsito utilizados normalmente (*Waze* ou *Google Maps*), onde é necessário abrir

o aplicativo para poder atualizar os dados ou obtê-los com os valores atualizados.

Com isso, não é necessário aplicar filtros e buscar a aceleração que possivelmente poderia estar com algum erro e ser necessário fazer tratamentos para evitar dados errados. Desta forma, como essa biblioteca utiliza modelos prontos e validados, e aplica nos dados *machine learning*, há muito mais confiança no resultado obtido. Além disso, como citado anteriormente, ela retorna um valor de confiabilidade, que se atualiza conforme o passar do tempo de análise. Com este valor é possível obter uma melhor avaliação do que está acontecendo, e assim evitar com que haja detecções erradas, como por exemplo, uma pessoa correndo ser detectada como andando em um veículo.

Na próxima Subseção será apresentada a forma de como é feita a atualização do *status* de um veículo para informar se o mesmo está estacionado ou não, utilizando esta biblioteca de desenvolvimento apresentada.

4.1.1.3 Obtenção do status do usuário

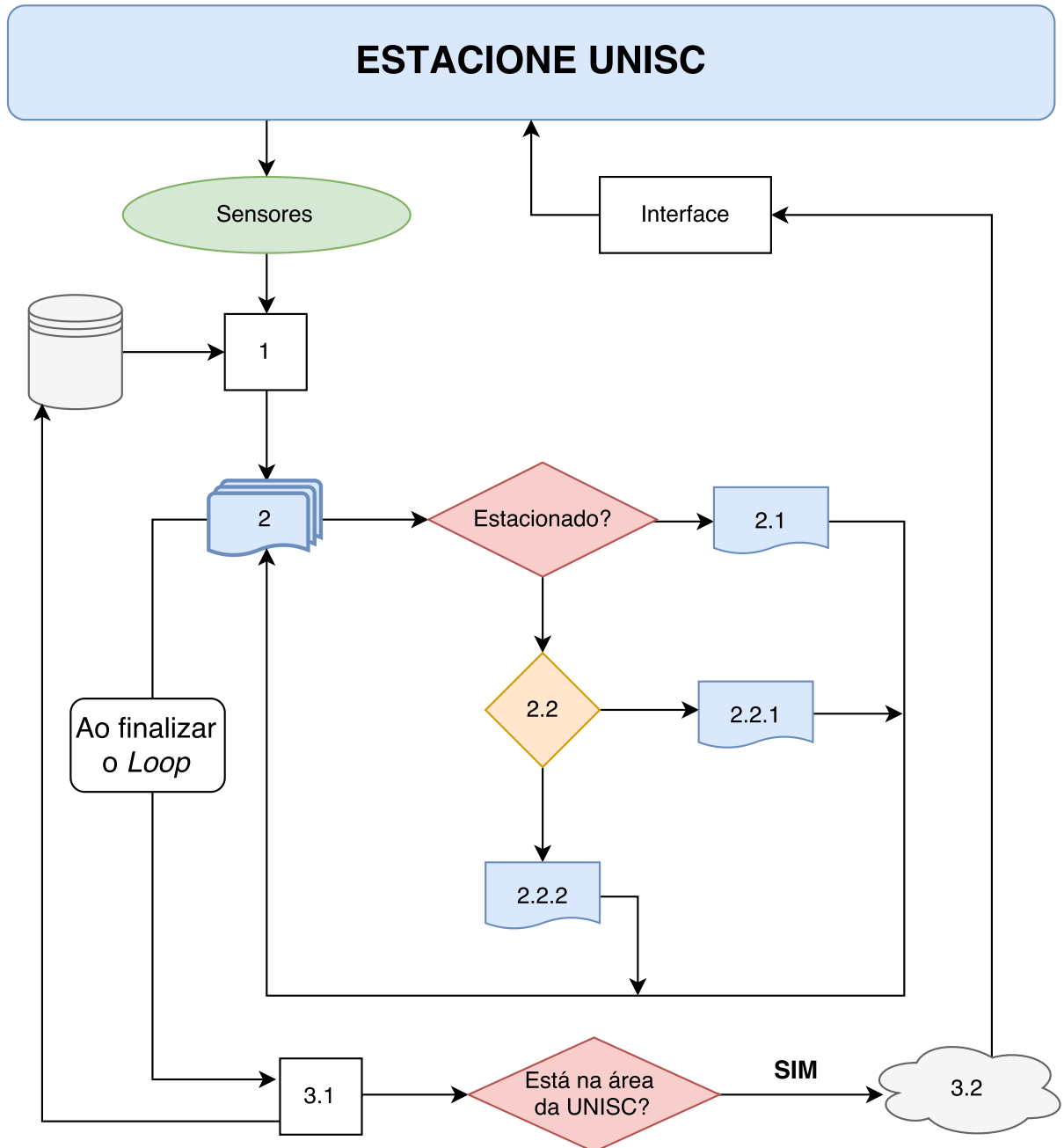
A partir da obtenção da forma na qual o usuário está se movimentando com informações provindas do seu aparelho móvel, lidas e retornadas pela *ActivityRecognitionAPI*, foi necessário analisar estas informações e aplicar mais alguns filtros para evitar que apenas esta API seja a responsável por determinar se o mesmo encontra-se estacionado ou não. Desta forma, a biblioteca atua como um auxílio na obtenção do tipo de deslocamento do usuário.

Para isso, foi criada uma base de dados interna que mantém as informações do último estado do dispositivo. A base de dados é criada quando o aplicativo é instalado no dispositivo e baseada em *SQLite*. Esta base de dados possui apenas uma tabela com informação de latitude, longitude e último estado do dispositivo, 0 (para não estacionado) e 1 (para estacionado).

A obtenção do estado atual do dispositivo é feita nos seguintes passos descritos, e também desenhados no fluograma da Figura 6:

- 1 Inicialmente são lidos os sensores de localização para obter a localização atual e velocidade, lê do banco de dados o estado atual do dispositivo e é feita a chamada para buscar o grau de confiança de cada atividade retornado pela *ActivityRecognitionAPI*;
- 2 Para cada atividade reconhecida, é verificado o último estado do dispositivo, se ele for estacionado, então realiza operações para detectar quando ele volta a se movimentar como um veículo:
 - 2.1 No caso de estar estacionado, é verificado se a atividade é **veículo (*In Vehicle*)**, se ela possui mais de 70% de confiabilidade e se a velocidade atual é maior que 15km/h. Se for, então salva em uma variável o estado atual do dispositivo para 0 (não estacionado);
 - 2.2 Caso não esteja estacionado, procura detectar o momento em que o dispositivo do

Figura 6 – Fluxograma da obtenção do estado atual do dispositivo



Fonte: Elaborado pelo autor.

usuário tem atividade relacionada a estar caminhando ou estar parado em repouso:

2.2.1 No caso da atividade ser **a pé (*On Foot ou Tiling*)**, se ela possui mais de 70% de confiabilidade e se a velocidade atual é menor que 10km/h, salva em uma variável o estado atual do dispositivo para 1 (estacionado);

2.2.2 No caso da atividade ser **em repouso (*Still*)**, se também possuir mais de 70% de confiabilidade e se a velocidade atual é menor que 2m/s, salva em uma variável o estado atual do dispositivo para 1 (estacionado);

3 Após realizar estas verificações para deixar a detecção mais robusta, antes de enviar ao servidor, ainda são feitos alguns testes. O principal, com objetivo de evitar o envio de informações iguais seguidamente ao servidor, é de que o novo estado detectado necessita ser diferente do atual que já está salvo no banco de dados interno:

3.1 Caso seja diferente, então altera o estado atual no banco de dados com o dado na variável auxiliar, juntamente com a localização onde o dispositivo se encontra atualmente;

3.2 Após, verifica se o dispositivo está dentro da área da UNISC, como exibido na Figura 7, e então envia a atualização para o servidor. Servidor este que retorna a informação com todos os pontos onde há veículos estacionados para ser atualizada no mapa.

Como pode ser visto nos itens descritos anteriormente, o grande diferencial desta ferramenta é conseguir fazer a detecção da atividade do usuário, porém mesmo obtendo esta informação, ainda foram aplicados mais alguns testes com objetivo de evitar erros nesta detecção.

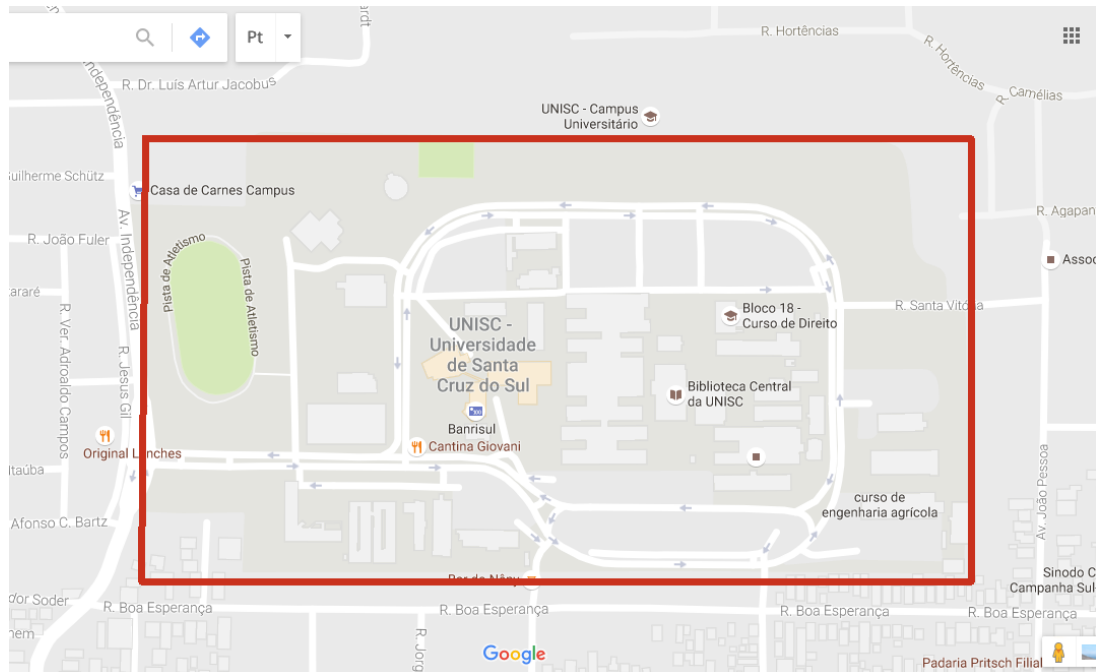
A comunicação com o servidor é feita diretamente por um gatilho enviado no momento em que há a detecção correta e atualização do banco de dados interno do dispositivo, e então é feito o envio de uma chamada *POST* com os dados do dispositivo. Na próxima Subseção será descrita estrutura de dados utilizada para o envio dos dados ao servidor e a retornada por ele para exibir os dados no mapa do aplicativo.

4.1.1.4 Envio e retorno do servidor

As informações trocadas pelo aplicativo e o servidor, hospedado na UNISC junto ao Departamento de Computação, são basicamente *strings*, dados de texto em formato *JSON*, com dados semelhantes. A diferença basicamente é que no envio ao servidor são enviadas apenas dados do dispositivo em questão, já na resposta do servidor todos os dados de cada dispositivo armazenados nele, são retornados com o objetivo de serem exibidos no mapa.

No exemplo apresentado no trecho de código Código 4.2, está sendo mostrado um *JSON* com os dados enviados ao servidor pelo dispositivo móvel. Como é possível de visualizar, são enviados os dados de localização, o seu estado, no caso do exemplo o ele estaria estacionado, e

Figura 7 – Área em que é feita a detecção de estacionamentos



Fonte: Elaborado pelo autor.

mais dois campos *mac_addr* e *action*.

O primeiro representa o *MAC Address* da placa de rede *Wi-Fi* do dispositivo, identificador único de cada interface de rede, que é utilizado como chave primária para consultas no banco de dados, e o segundo é a ação que será tomada no servidor, sendo 0 (zero) para apenas leitura dos dados e 1 (um) para atualização. Desta forma, se o usuário apenas quer atualizar o mapa, é enviado este arquivo com o campo *action* com valor 0 (zero). Já, no momento em que o dispositivo detecta que o usuário está estacionado, ele faz o envio com este valor sendo 1 (um).

Código 4.2 – Exemplo do *JSON* enviado ao servidor

```
{
  "mac_addr": "XX:XX:XX:XX:XX:XX",
  "latitude": "-29.696484",
  "longitude": "-52.435429",
  "status": "1",
  "action": "1"
}
```

Na resposta enviada pelo servidor para o dispositivo, há muito mais informações. Cada grupo de campos corresponde a um dispositivo que está salvo no servidor e possui *status* com o valor 1 (estacionado). No retorno do servidor, apenas os dispositivos salvos que possuam *status* de estacionados é que são retornados pela consulta e enviados para o dispositivo que fez a requisição. Por exemplo, se houver cento e cinquenta carros salvos, mas apenas cem carros estacionados na UNISC que estejam com *status* de estacionados, o *JSON* conterá apenas cem

grupos de dados, que correspondem aos que estão estacionados, e assim por diante. O trecho de código Código 4.3 exemplifica esta descrição.

Código 4.3 – Exemplo do *JSON* retornado pelo servidor

```
{
  {
    "latitude":"-29.696857",
    "longitude":"-52.435387",
    "status":"1",
    "data_hora":"2016-11-10 23:42:49"
  },
  {
    "latitude":"-29.696466",
    "longitude":"-52.435680",
    "status":"1",
    "data_hora":"2016-11-10 23:39:11"
  },
  {
    "latitude":"-29.696596",
    "longitude":"-52.435532",
    "status":"1",
    "data_hora":"2016-11-11 00:19:36"
  }
  ...
}
```

Como apresentado nesta Subseção, a maneira de comunicação cliente-servidor é simples e sem dados específicos, apenas uma chamada *POST* para enviar ou ler os dados. Além do fato de ser simples, acaba sendo eficaz com conexões que possuem sinal ruim, por exemplo, internet móvel. Desta forma, mesmo utilizando plano de dados do dispositivo móvel, o aplicativo conseguirá se comunicar com sucesso pelo fato do baixo uso de rede necessário para realizar as consultas.

Na próxima Subseção, será apresentada a forma com a qual os dados são exibidos ao usuário, basicamente a interface do aplicativo em si. Um mapa com os dados de veículos estacionados para que o usuário possa interpretar e decidir onde há a maior probabilidade de encontrar vaga para estacionar.

4.1.1.5 Exibição dos dados

A maneira de como os dados são apresentados ao usuário sempre deve fornecer a informação da maneira mais clara possível, portanto muitas vezes uma interface simples e amigável pode trazer melhores resultados do que uma cheia de campos e informações que acabam confundindo o usuário ao invés de ajudá-lo.

Com este pensamento, foi aplicado à interface do aplicativo Estacione UNISC apenas um mapa que utiliza a API do *Google Maps* (GOOGLE DEVELOPERS, 2016e). A partir dele, a maneira inicial pensada foi de exibir os dados de forma simples, com apenas os marcadores para cada veículo estacionado. O grande problema desta abordagem é de que há um grande amaranhado de informação e vários marcadores, um sobre o outro, o que torna difícil a compreensão do usuário.

A proposta do trabalho apresentava outra maneira, mais eficaz, porém ainda confusa. Os chamados mapas de calor. O grande problema deles, é que não conseguem transmitir informação ao usuário. Além de que conforme o *zoom* que o usuário aplica, o mapa vai atualizando, e as cores mudam, o que faz com que o mapa não seja constante. Por exemplo, com 100% de *zoom*, um círculo vermelho poderia aparecer sobre a UNISC, pelo fato de ter vários veículos estacionados. Porém ao mudar o *zoom* para 25%, tudo o que era vermelho, fica verde, pois há a separação dos pontos, o que faz com que a percepção mude totalmente.

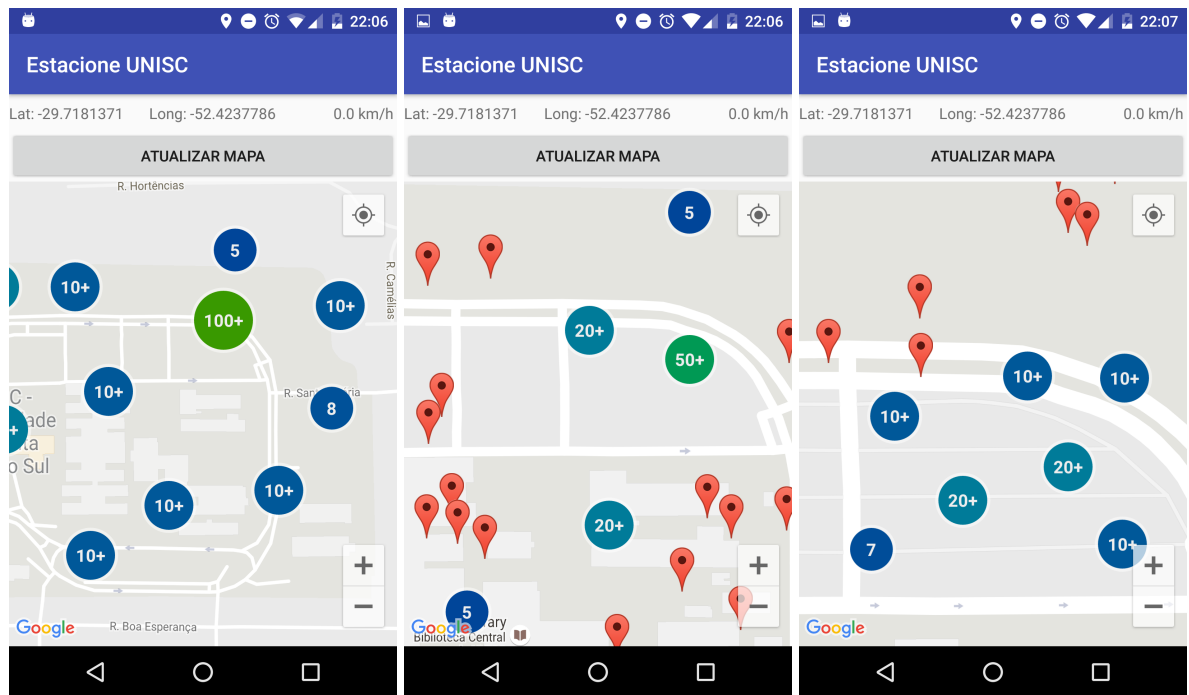
Desta forma, foi encontrada uma maneira de conseguir apresentar os dados de forma quantitativa, baseada na primeira ideia de exibir todos os marcadores, porém com uma técnica de *clustering* (agrupamento) de marcadores por proximidade. Técnica esta que pode ser utilizada graças a um utilitário do *Google Maps*, chamado de *Marker Clustering* (GOOGLE DEVELOPERS, 2016d).

Com o agrupamento dos marcadores, o mapa apresenta um número aproximado de quantos pontos estão naquela determinada área, no caso do aplicativo Estacione UNISC, busca exibir o número aproximado de veículos estacionados naquela área. Ao alterar o *zoom*, o mapa é atualizado para os marcadores cuja distância é maior que o máximo permitido para aquele centro de *cluster* serem mostrados separados, enquanto que o resto dos marcadores permanecem agrupados. O *zoom* possui 5 níveis, de 1 a 5, sendo o primeiro o *zoom* mais distante e o segundo o mais aproximado.

As sub figuras (a), (b) e (c) exibidas na Figura 8 mostram a diferença dos *clusters* conforme o *zoom* aplicado. Na primeira, o *zoom* possui nível 2, desta maneira a imagem contempla mais território, ao contrário das outras duas imagens. A sub figura (b) possui *zoom* nível 3, ou seja um nível médio, em que é possível visualizar com mais detalhamento os dados, já na terceira sub figura, é possível ver os dados mais aproximados, já que esta possui *zoom* nível 4. Assim, é possível ter uma visão melhor dos estacionamento, onde é possível visualizar a separação em grupos de marcadores menores e alguns marcadores separados de grupos, além das

pequenas ruas que cruzam os estacionamentos.¹

Figura 8 – Exemplo de funcionamento do agrupamento de marcadores com níveis de zoom diferentes



(a) Zoom nível 2

(b) Zoom nível 3

(c) Zoom nível 4

Fonte: Elaborado pelo autor.

Para a criação destes grupos de marcadores, utiliza-se o *JSON* de resposta do servidor, e se percorre buscando cada um dos pontos (localização dos dispositivos) retornados, os quais são utilizados para criar marcadores individuais e estes são enviados para a função que faz o agrupamentos dos mesmos e os atualiza a cada troca de zoom. Além disso, a cada chamada para o servidor em que há atualização dos dados, os *clusters* são removidos e gerados novamente, pois pode acontecer de algum veículo ter saído do estacionamento e outros podem ter estacionado, com isso os dados permanecem sempre atualizados.

4.1.2 Servidor

O funcionamento do aplicativo depende de um servidor onde as informações providas pela aplicação são armazenadas e consultadas quando necessário. Desta forma além do desenvolvimento da solução, foi necessário desenvolver a parte de *back end* para armazenar as informações do preenchimento dos estacionamentos e estas serem possíveis de serem acessadas pelo aplicativo de qualquer lugar, desde que o mesmo possua acesso à internet.

¹Os dados utilizados nestes exemplos não são reais. Foram dados inseridos manualmente nos testes realizados durante o desenvolvimento.

4.1.2.1 Back end e base de dados

O servidor está instalado no Departamento de Computação da UNISC, e é acessado por uma chamada enviada pelo aplicativo quando necessário, normalmente quando há requisição para atualização de dados ou para a leitura dos mesmos. Como descrito anteriormente, a comunicação entre o aplicativo e o servidor é simples. Há pequenas trocas de mensagens em formato de texto com os dados de preenchimento, provindos do servidor, e dados de localização e *status* enviados pelo aplicativo.

O servidor, desenvolvido em linguagem *PHP*, possui também conexão com um banco de dados *MySQL* onde armazena os dados em duas tabelas. Uma para o estado atual dos estacionamentos, e outra para armazenar o histórico, dessa maneira qualquer alteração no estado dos dispositivos é atualizado e fica salvo. Assim é possível de se obter, após alguns meses de uso e atualização, possíveis curvas de preenchimento durante determinados dias e horários. A Figura 9 exibe o *schema* do banco de dados do servidor, com a descrição de cada uma das tabelas, juntamente com seus campos e respectivos tipos de dados.

Figura 9 – Tabelas do banco de dados para armazenar informações de preenchimento dos estacionamentos

estacionamentos		historico_estacionamentos	
PK	mac_addr [<i>varchar(12)</i>]		latitude [<i>decimal(10, 6)</i>]
	latitude [<i>decimal(10, 6)</i>]		longitude [<i>decimal(10, 6)</i>]
	longitude [<i>decimal(10, 6)</i>]		status [<i>tinyint(1)</i>]
	status [<i>tinyint(1)</i>]		data_hora [<i>datetime</i>]
	data_hora [<i>datetime</i>]		

Fonte: Elaborado pelo autor.

Como o tempo de testes foi curto e praticamente a maioria dos testes foram feitos durante o desenvolvimento sem envolver pessoas que frequentam a UNISC, não há uma informação na tabela de histórico que possa gerar um gráfico exibindo o preenchimento em determinado período.

4.1.2.2 Atualização dos dados

A atualização dos dados é executada conforme a chamada enviada pelo aplicativo. Dentro dos parâmetros enviados pela aplicação, há um em específico que indica se haverá atualização ou apenas leitura dos dados, o parâmetro *action*. Caso possuir valor 0 (zero) apenas lê e re-

torna os dados, se possuir valor 1 (um), então atualiza e retorna os dados atualizados para o dispositivo.

Esta foi uma forma simples de implementar uma leitura sem ser necessário fazer duas chamadas diferentes para o servidor, o que aumentaria o tráfego de informações. Ao iniciar o aplicativo, é feita uma consulta de apenas leitura, obtendo os dados atuais. A cada alteração no estado do dispositivo, é enviado uma atualização com os novos dados, tanto para a tabela de estado atual, quanto para a de histórico, pois houve uma mudança de estado e todas são salvas. Além disso, se o usuário desejar atualizar o mapa, pressionando o botão "Atualizar Mapa", novamente é feita apenas uma consulta, sem alterar nenhum dado no banco de dados.

O retorno dos dados é feito todo enviando uma *string* em formato *JSON*, o faz a comunicação praticamente não realizar tráfego de dados. Pois são apenas algumas informações de texto, assim como uma mensagem enviada por algum aplicativo de mensagens instantâneas. Um exemplo dos dados, em formato *JSON*, retornados pelo servidor podem ser vistos no trecho de código Código 4.2.

4.2 Validação e Testes

Os testes do sistema foram planejados para abranger todo o período de desenvolvimento, pois para cada passo era necessário realizar testes e validar os dados para ter certeza que tudo estava sendo feito corretamente. Assim que foi finalizada a primeira parte do desenvolvimento, relacionada à obtenção do dispositivo já estava sendo feitos os testes de desenvolvimento e funcionais.

Assim que finalizada a parte inicial de busca a localização do dispositivo, iniciou-se o desenvolvimento da parte de detecção da forma de movimentação, juntamente com a definição do status do dispositivo, e os testes foram iniciados da mesma forma. Como descrito na Seção 4, houve a tentativa de implementar uma maneira de ler a aceleração do dispositivo, e com ela foi necessário realizar testes que verificassem se esta implementação estaria com valores corretos. A partir dos testes foi visto que não seria possível encontrar a aceleração devido à forma de como o acelerômetro funciona.

Desta maneira, com a mudança para o uso da API de detecção de atividade, também foi necessário testá-la e verificar realmente a lealdade da mesma. Após implementada, foi testada a partir de corridas e caminhadas, ou mesmo ao andar em um veículo. Não seria possível de se emular uma movimentação de carro em um celular, por exemplo, desta maneira os testes

tiveram que ser feitos manualmente através do uso de um carro para verificar a confiança da API.

Com estes testes realizados, foi visto que apesar dos resultados retornados por ela serem muito precisos, ainda assim há variações aleatórias conforme o decorrer da movimentação. Através desta análise, que foi decidido aplicar outros filtros ligados à velocidade ao retorno desta API de detecção de movimento, com intuito de tornar os resultados os mais próximos da realidade.

Após os testes finalizados e a parte da detecção finalizada, foram feitos os testes relativos à detecção do status do dispositivo, se estivesse estacionado ou não. O que necessitou de testes manuais, utilizando um carro para fazer a validação e encontrar possíveis erros na execução do algoritmo. Além destes testes, foi realizada a validação dos dados exibidos no mapa, que foram satisfatórios desde o início, sempre sendo apresentados da maneira correta. Assim como testes relacionados à comunicação com o servidor e tratamento dos dados pelo mesmo, que durante a implementação foram testados localmente, e assim que disponíveis no servidor do Departamento de Computação da UNISC, já estavam validados e funcionando corretamente.

O aplicativo final não chegou a ser utilizado e testado por outros usuários, apenas por usuários convidados que foram auxiliares nos testes reportando problemas, principalmente em relação à detecção do *status*. Apesar de nenhum teste externo e análises sobre os dados de usuários serem feitos, a detecção de status, comunicação e atualização dos dados com o servidor estão funcionando corretamente. A grande questão encontrada foi a imprecisão do GPS em determinados momentos, o que pode gerar dados de posicionamento incorretos.

Como o aplicativo não foi liberado para uso externo, apenas durante testes, não é possível apresentar dados reais, com opiniões, dados quantitativos e qualitativos. Os dados existentes são apenas referentes a testes locais realizados e em pequena escala, que não representam a realidade. Desta maneira, o trabalho não entrega um aplicativo finalizado, que necessita obter dados e opiniões de usuários para serem analisados. Porém, foi entregue uma estrutura que já possui seu alicerce para construir mais ferramentas a partir dela, com uma base de dados para receber dados de posicionamentos e armazenamento de histórico de localização.

Neste capítulo foi apresentada a solução desenvolvida pelo trabalho, com o detalhamento do aplicativo e do servidor, desde o funcionamento até a comunicação entre ambos, além de apresentar os testes e validações realizadas. No próximo capítulo são apresentados os resultados e as considerações finais do trabalho.

5 CONSIDERAÇÕES FINAIS

O uso de tecnologias na área de mobilidade urbana têm se mostrado muito útil, principalmente com a utilização de câmeras de monitoramento. Porém ainda é preciso evoluir mais e estudar novas maneiras de implementar melhorias como controle de estacionamentos e de tráfego com objetivo de evitar congestionamentos e falta de lugares para estacionar.

O trabalho e a pesquisa realizada demonstram que é possível desenvolver sistemas que ajudem neste controle, porém é necessário aprofundar as pesquisas e procurar agregar diferentes formas de monitoramento, além de colher dados de diferentes formas para obter a melhor informação possível, como por exemplo obter o estados de vagas de estacionamentos através de câmeras e informar ao usuário através de um mapa dentro de um aplicativo. Além disso, o presente trabalho tem foco em estacionamentos abertos, já que em estacionamentos cobertos ou locais fechados o sinal de GPS perde precisão, além do fato de já existirem diferentes ferramentas para monitoramento em locais fechados.

Os resultados obtidos foram satisfatórios na medida do possível, pois o objetivo era obter uma versão estável do aplicativo e liberar para uso dos alunos e professores da UNISC. Porém, mesmo com a liberação da versão, um mês de utilização do aplicativo não seria suficiente para encontrar um padrão em que é possível dizer que o resultado foi igual ao esperado. O grande fator que está atrelado é a adesão dos usuários, primeiro pois muitos frequentadores da UNISC acabam indo de ônibus, de carona ou a pé para a universidade, e segunda porque muitos possuem aparelhos cujo aplicativo não é compatível.

Apesar disso, os testes realizados internamente durante o desenvolvimento se mostraram eficientes, onde foi possível detectar a forma de movimento do dispositivo, e caso fosse veículo, detectar o seu *status* de estacionado ou não. As dificuldades e problemas encontrados durante o processo de desenvolvimento foram, principalmente, relacionados ao GPS. Há uma grande variação nos dados de geolocalização, principalmente se conectado à uma rede de dados móvel. A posição nem sempre é a real, pois em muitos casos de testes o posicionamento estava longe do local de onde o dispositivo realmente estava ou então abrangendo uma circunferência de cerca de 100 metros raio, o que acaba levando o aplicativo a ser impreciso tanto na posição quanto na determinação do *status* do veículo.

Apesar do fato do GPS possuir certa imprecisão ao fornecer os dados de localização, não chega-se a conclusão de inviabilizar o trabalho, visto que ele possui grande potencial, principalmente se agregadas novas maneiras de monitoramento, como por exemplo, mapear as áreas de

estacionamento e verificar quando a posição do dispositivo estivesse sobre uma área mapeada como estacionamento, para assim detectar o *status* de estacionado.

Outras dificuldades durante o processo também foram encontradas, principalmente em relação ao desenvolvimento pelo fato de ser uma plataforma nova que nunca havia-se trabalhado. A detecção de atividade do usuário foi uma das principais, pois foi feito um grande esforço em buscar ler os dados diretamente do dispositivo sem sucesso, até a API de reconhecimento de movimento ser encontrada e aplicada, o que facilitou este passo do trabalho, apesar de também ter seus problemas no início, até estar 100% funcional. API esta que se mostrou muito útil pelo fato de trabalhar em *background* e lendo os dados diretamente na camada mais baixa do sistema melhorando o desempenho e devolvendo para o aplicativo o resultado da análise. Ela se mostrou ser muito útil, e pode se aplicada em diferentes tipos de projetos que dependam da análise e detecção de atividade e de movimento do dispositivo.

Apesar de uma versão estável não ter sido publicada, a contribuição e objetivos principais do trabalho foram realizadas, que eram de obter uma estrutura e uma ferramenta para ajudar na detecção de movimento do usuário e buscar identificar quando o mesmo está estacionado, salvar a informação e torná-la disponível num servidor. Estrutura esta que pode ser utilizada para diversas outras áreas, como obter e salvar dados de outras atividades que possuam deslocamentos e utilizam aplicativos para fazer controle de algo relacionado à mudança de posicionamento, também podem aproveitar a estrutura.

Em relação a trabalhos futuros, podem-se citar a aplicação de melhorias no trabalho atual, como adesão de outras formas de monitoramento, como obter informações dos pedestres, por onde andam e qual o local que mais frequentam, identificando assim o perfil de trânsito da universidade. Ou então, ter um aplicativo para controle de atividade física de pessoas, que utilize a parte desenvolvida da API de detecção de movimento e salve os dados de deslocamento na base de dados, obtendo o tempo que caminhou e a distância total, que são enviados ao médico para verificar se condiz com o tratamento.

Assim como o trabalho citado acima, é possível se pensar em um trabalho futuro que também aplica melhorias e agrega novas tecnologias e formas de monitoramentos ao trabalho atual, no qual são mapeadas todas as vagas de um local obtendo a geolocalização das mesmas. A partir de um raio definido por padrão, é possível de se analisar quando o veículo está sobre a área da vaga, determinando que ele está estacionado, se permanecer por algum determinado tempo mínimo, e no momento em que não há mais a intersecção entre a posição do dispositivo

e da área da vaga, significa que ele não está mais estacionado.

Outro trabalho que pode ser desenvolvido baseado neste base de dados é indicar a probabilidade de estacionamento através de GPS e processamento de imagens. Através de imagens de câmeras, obter o status de um estacionamento e salvar as informações na base de dados e a partir disso o usuário já fica sabendo o estado daquela área, e através dos dados de GPS o aplicativo fornece o tempo até o destino e um possível horário máximo até o qual o usuário possa sair do local atual para ainda encontrar vagas, baseado no histórico de preenchimento daquela área, que é fornecido pelos dados das câmeras.

A conclusão final que se chega com o trabalho é que é necessária muita pesquisa ainda para conseguir obter melhores resultados, porém com pequenos passos e pequenas aplicações e análises é possível buscar soluções para isso. Este trabalho mostra que apesar de depender de GPS, cujo sinal é muito inconstante, e da adesão dos usuários, a ferramenta tem uma grande utilidade e que, como citado anteriormente, se for aliada a outras tecnologias e projetos que podem ajudar cada vez mais este monitoramento da mobilidade urbana. O futuro na área de monitoramentos urbanos aliadas à computação é muito promissor, principalmente com a conectividade atual, que tende a melhorar. Isto faz com que seja possível de se observar que a união entre tecnologias, áreas de atuação e de pesquisa trazem grande benefício para as cidades e conseqüentemente para as pessoas.

REFERÊNCIAS

- ARNOTT, R.; INCI, E. An integrated model of downtown parking and traffic congestion. *Boston College Department of Economics*, n. 608, fev. 2005. Disponível em: <<https://ideas.repec.org/p/boc/bocoec/608.html>>. Acesso em: 25 out. 2016.
- DENATRAN. Departamento Nacional de Trânsito. *Código de Trânsito Brasileiro*. 2008. Disponível em: <http://www.denatran.gov.br/publicacoes/download/ctb_e_legislacao_complementar.pdf>. Acesso em: 03 mai. 2016.
- GOOGLE INC. Google Developers. *Activity Recognition: Use low power sensors to detect user activity*. 2015. Disponível em: <<https://www.youtube.com/watch?v=S8sugXgUVEI>>. Acesso em: 12 set. 2016.
- GOOGLE INC. Google Developers. *ActivityRecognitionApi*. 2016. Disponível em: <<https://developers.google.com/android/reference/com/google/android/gms/location/ActivityRecognitionApi>>. Acesso em: 21 set. 2016.
- GOOGLE INC. Google Developers. *Android 6.0 Marshmallow - APIs*. 2016. Disponível em: <<https://developer.android.com/about/versions/marshmallow/android-6.0.html>>. Acesso em: 10 mai. 2016.
- GOOGLE INC. Google Developers. *Dashboards - Platform Versions*. 2016. Disponível em: <<https://developer.android.com/about/dashboards/index.html>>. Acesso em: 15 Nov. 2016.
- GOOGLE INC. Google Developers. *Google Maps Android Marker Clustering Utility*. 2016. Disponível em: <<https://developers.google.com/maps/documentation/android-api/utility/marker-clustering>>. Acesso em: 02 Nov. 2016.
- GOOGLE INC. Google Developers. *Google Maps API*. 2016. Disponível em: <<https://developers.google.com/maps/>>. Acesso em: 02 mai. 2016.
- GSM ARENA. *LG Nexus 5 - Full phone specifications*. 2015. Disponível em: <http://www.gsmarena.com/lg_nexus_5-5705.php>. Acesso em: 13 nov. 2016.
- GUO, L. et al. Dostra: Discovering common behaviors of objects using the duration of staying on each location of trajectories. 2015. Disponível em: <<http://www.aaai.org/ocs/index.php/WS/AAAIW15/paper/view/10135/10247>>. Acesso em: 01 mar. 2016.
- HUANG, G. et al. Discovery of stop regions for understanding repeat travel behaviors of moving objects. *Journal of Computer and System Sciences*, v. 82, n. 4, p. 582 – 593, 2016. ISSN 0022-0000. Trajectory-based Behaviour Analytics. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0022000015001348>>. Acesso em: 01 mar. 2016.
- IBGE. Instituto Brasileiro de Geografia e Estatística. *Noções Básicas de Cartografia*. 1987. Disponível em: <http://www.ibge.gov.br/home/geociencias/cartografia/manual_nocoos/representacao.html>. Acesso em: 03 mai. 2016.
- JAYASYDHA, N.; KARUNAMOORTHY, B.; SURESHKUMAR, R. Design and implementation of a system for image based automatic detection and counting of vehicles. In: . [S.l.: s.n.], 2015. v. 10, n. 20, p. 16963 – 16968.

LIU, J.; MOHANDES, M.; DERICHE, M. A multi-classifier image based vacant parking detection system. In: *Electronics, Circuits, and Systems (ICECS), 2013 IEEE 20th International Conference on*. [S.l.: s.n.], 2013. p. 933–936.

LUZ, P. R. P. da. *Um estudo de caso utilizando técnicas de preferência declarada para análise do fluxo e permanência de veículos em áreas delimitadas*. Dissertação (Mestrado) — Universidade Federal de Santa Catarina (UFSC), Centro Tecnológico, Programa de Pós-Graduação em Engenharia de Produção, 1997. Disponível em: <<https://repositorio.ufsc.br/handle/123456789/77066>>. Acesso em: 25 out. 2016.

MARQUES, G. D. C. *Ensino de Física Online - e-física*. Instituto de Física - Universidade de São Paulo (USP), 2008. Disponível em: <<http://efisica.if.usp.br/mecanica>>. Acesso em: 09 mai. 2016.

MOSCARELLI, F. C. *Proposta de método para análise de demanda para um modo combinado de transporte associado a estacionamentos dissuasórios*. Dissertação (Mestrado) — Universidade Federal do Rio Grande do Sul (UFRGS), Escola de Engenharia, 2003. Disponível em: <<http://www.lume.ufrgs.br/handle/10183/18984>>. Acesso em: 12 nov. 2016.

NASA. National Aeronautics and Space Administration. *Global Positioning System (GPS)*. 2014. Disponível em: <<http://www.nasa.gov/directorates/heo/scan/communications/policy/GPS.html>>. Acesso em: 05 mai. 2016.

NCO. National Coordination Office for Space-Based Positioning. *GPS Standard Positioning Service (SPS) Performance Standard - 1st Edition*. [S.l.], 1993. Disponível em: <<http://www.gps.gov/technical/ps/1993-SPS-signal-specification.pdf>>. Acesso em: 05 mai. 2016.

NCO. National Coordination Office for Space-Based Positioning. *GPS Educational Poster*. 2013. Disponível em: <<http://www.gps.gov/multimedia/poster/>>. Acesso em: 05 mai. 2016.

NCO. National Coordination Office for Space-Based Positioning. *GPS Standard Positioning Service (SPS) Performance Standard*. 2016. Disponível em: <<http://www.gps.gov/technical/ps/>>. Acesso em: 05 mai. 2016.

NETO, E. C. et al. Development control parking access using techniques digital image processing and applied computational intelligence. *IEEE Latin America Transactions*, v. 13, n. 1, p. 272–276, Jan 2015. ISSN 1548-0992.

OHASHI, H. et al. Trip-extraction method based on characteristics of sensors and human-travel behavior for sensor-based travel survey. *Journal of Information Processing*, v. 24, n. 1, p. 39–48, 2016.

OS. Ordnance Survey. *A guide to coordinate systems in Great Britain*. 2015. Disponível em: <<https://www.ordnancesurvey.co.uk/docs/support/guide-coordinate-systems-great-britain.pdf>>. Acesso em: 07 mai. 2016.

PAIVA, M. *Implantação de estacionamentos de automóveis e bicicletas integrados ao transporte público*. Dissertação (Mestrado) — Instituto Militar de Engenharia, Curso de Mestrado em Engenharia de Transportes, 2008. Disponível em: <http://www.dominiopublico.gov.br/pesquisa/DetalheObraForm.do?select_action=&co_obra=116117>. Acesso em: 06 mar. 2016.

PALANDI, J. et al. *Cinemática e Dinâmica*. Departamento de Física - Universidade Federal de Santa Maria (UFSM), 2010. Disponível em: <<http://coral.ufsm.br/gef/Cadernos/CineDin.pdf>>. Acesso em: 09 mai. 2016.

RIBEIRO, S. K. et al. Transport and its infrastructure. In: METZ, B. et al. (Ed.). *Climate Change 2007: Mitigation of Climate Change*. Cambridge: Cambridge University Press, 2007. cap. 5, p. 323 – 385. Disponível em: <<https://www.ipcc.ch/pdf/assessment-report/ar4/wg3/ar4-wg3-chapter5.pdf>>. Acesso em: 27 out. 2016.

SILVA, G. R. L. *Sistema Multiagente para Simulação da Dinâmica de Estacionamentos – SMSDEs*. Dissertação (Mestrado) — Universidade Federal de Minas Gerais (UFMG), Curso de Pós-Graduação em Engenharia Elétrica, 2011. Disponível em: <http://www.ppgee.ufmg.br/diss_defesas_detalhes.php?aluno=85>. Acesso em: 13 mar. 2016.

VEIT, E. A. *Um estudo de Cinemática*. Instituto de Física - Universidade Federal do Rio Grande do Sul (UFRGS), 2007. Disponível em: <http://www.if.ufrgs.br/cref/ntef/cinematica/IGCin_texto.pdf>. Acesso em: 09 mai. 2016.

WANG, J. et al. Predictability of road traffic and congestion in urban areas. *PLoS ONE*, Public Library of Science, v. 10, n. 4, p. 1–12, 04 2015. Disponível em: <<http://dx.doi.org/10.1371/journal.pone.0121825>>. Acesso em: 05 mar. 2016.

ZHANG, X. W.; YU, B. Design and implementation of university parking query system. In: *Mechatronics and Industrial Informatics II*. [S.l.]: Trans Tech Publications, 2014. (Applied Mechanics and Materials, v. 596), p. 909–913.

ZHENG, Y. et al. Understanding transportation modes based on gps data for web applications. *ACM Trans. Web*, ACM, New York, NY, USA, v. 4, n. 1, p. 1:1–1:36, jan. 2010. ISSN 1559-1131. Disponível em: <<http://doi.acm.org/10.1145/1658373.1658374>>. Acesso em: 08 mar. 2016.